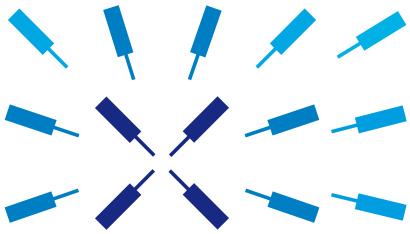
HF2 User Manual



Zurich
Instruments

HF2 User Manual

Zurich Instruments AG

Publication date Revision 30817

Copyright © 2008-2015 Zurich Instruments AG

The contents of this document are provided by Zurich Instruments AG (ZI), “as is”. ZI makes no representations or warranties with respect to the accuracy or completeness of the contents of this publication and reserves the right to make changes to specifications and product descriptions at any time without notice.

LabVIEW is a registered trademark of National Instruments Inc. All other trademarks are the property of their respective owners.

Revision History

Revision 30817, 19-Jun-2015:

Document maintenance - editorial updates.

Revision 26211, 30-Sep-2014:

Document maintenance - editorial updates.

- Sweeper: Sinc filter speed improvement
- Improved locking range for PLL and Extref mode (external reference)
- Up to data LabOne APIs

Revision 23220, 24-Apr-2014:

As of software release 14.02, Zurich Instruments combines the installers for the HF2 Series and UHF Series Instruments. Hence the installation of ziBase is superseded by the installation of LabOne package. Further, all programmer's reference has been moved to a separated LabOne Programming Manual document as Zurich Instruments provide one API that suits both HF2 and UHF users. The following sections of this document have been updated:

- The chapter on instrument programming has moved to the LabOne Programming Manual
- Updated getting started section with LabOne installation instructions
- New feature (HF2LI without HF2LI-option): added support for demodulation using second oscillator
- New feature (HF2LI-PID): added support for PID setpoint sweep
- New feature (Sweeper): added phase unwrap for polar coordinate display
- Specifications: official support for Windows 8
- Specifications: modified initial oscillator accuracy of HF2LI-UHS and HF2IS-UHS options from ±0.15 ppm to ±0.5 ppm
- Added section regarding location of log files in the Troubleshooting section
- Several minor editorial edits

Revision 15296, 20-Dec-2012:

Document update for 12.08 software release:

- Specifications: changed the AC range from 3.5 V to 3.3 V, and introduced AC signal limitation when AC coupling is used (max 0.6 V)
- Specifications: 100 V supply systems (Japan) require external transformator - updated specification and getting started
- Specifications: added performance diagram and test specifications for lock time and dynamic reserve
- HF2PLL: increased PLL gain for low quality factors (Q = 4 to 10) by up to factor 100, increased the range of PLL time constant parameter

- HF2LI-PID option: revised functional description, improved and expanded tutorials, added recipe to find PID parameters manually
- HF2LI-RT option: added several examples
- HF2IS: added description of impedance measurement with Nyquist plot in HF2TA section
- Added LabVIEW VIs to perform Q-Control and Tip Protection
- Added status indicator for supply voltage in Connectivity tab

Revision 13029, 26-Jul-2012:

Important document update for 12.02 software release:

- added support for arbitrary input unit scaling for HF2LI/HF2PLL/HF2IS
- added integrated preamplifier unit conversion for HF2LI/HF2PLL/HF2IS
- added flexible lock-in filter dynamics setting (BW, NEPBW, effective TC, TC per order)
- reworked sweeper with multi-quantity support (frequency, phase, amplitude, offset), reference sweeps, calibration sweeps, persistent sweeps, ping pong sweeps, Nyquist plot function, HF2PLL Peak Analyzer (Q-Factor calculator)
- reworked spectroscope with frequency analysis support
- moved noise analysis tools from numerical tab to FFT spectrum analyzer
- reworked FFT spectrum analyzer with windowing, new operation mode (continuous and block), and 2nd cursor
- PID option inputs: added many features, new input units (frequency, Theta, others), PID Advisor (closed and open simulation, several DUT models), PID auto tuner, PID setpoint toggle, external PID setpoint, dual frequency resonance tracking (DRFT), cascaded PID
- PID option outputs: added feature to output several complex quantities on auxiliary outputs, e.g. PID sideband analyzer, PID output (e.g. dissipation monitor dA), PID default output value
- PLL option: improved HF2PLL Advisor
- RT option: added named register, register formats (hex, int, float32), load and save of register configuration, improved performance of data function, reorganized chapter
- added description for Python and MATLAB to programming chapter
- reorganized programming and API descriptions
- added support for Sync TTL output on DIO 0/1 connectors
- updated tutorial chapter, added new tutorial for HF2LI-PID Quad-PID Controller option
- revised and improved clarity of the specification chapter, added performance diagrams and test specifications
- updated section on signal processing basics with settling time, filter characteristics
- HF2TA: added applications and performance test specifications
- added recommendation for preamplifier cable ordering
- added quick links for customer support
- revised all graphical user interface panels

Revision 7921, 26-May-2011:

Important document update for 11.02 software release:

- added Zoom FFT / FFT spectrum analyzer panel
- added Sinc filter to eliminate omega and 2-omega components to HF2LI
- added description of new HF2LI-PID option
- added support for rectangular output waveform (Signal Outputs)
- improved frequency sweeper panel interface
- added browsable history of the command log
- added output current specification
- added new tutorials first time HF2LI user
- performance: increased cumulated demodulator sample rate to up to 700kS/s
- RT option: increased RT processor speed by factor of 4, improved graphical user interface, increased the number of available user registers to 64
- PLL option: added 2-omega PLL mode, improved PLL Advisor

Revision 5991, 22-Dec-2010:

Important document update for 10.06 software release:

- added specification of full range input sensitivity
- added recalibration requirements, improved oscilloscope
- updated lock-in triggering section
- added documentation on noise measurement tool
- added host computer hardware requirements
- many updates in the functional descriptions
- added node definition chapter
- updated RT and API chapters
- updated software installation section
- added output phase noise specification
- clarification of NEP bandwidth
- added AM/FM modulation and PLL tutorials
- added HF2TA datasheet
- added Linux installation requirements
- refined internal oscillator specifications
- added documentation of PLL Advisor
- added description of the noise equivalent power bandwidth (NEPBW)
- added description on SSH port forwarding

Revision 4502, 16-Jun-2010:

Major document revision for 10.03 software release:

- clarified the LabVIEW programming section

Revision 4368, 30-May-2010:

- PLL and AM/FM modulation options for HF2LI
- completely revised the functional description
- created 2 separated chapters for HF2IS/HF2LI graphical user interface descriptions
- added noise measurement and external reference tutorials
- exchanged all screenshots
- update of real-time section and all programming interfaces

- added maximum ratings table
- reduced auxiliary input samplig rate from 500 kS/s to 400 kS/s
- corrected internal oscillator accuracy (standard non-UHS specification)
- added detailed reference specifications

Revision 2670, 20-Oct-2009:

- dual-harmonic mode supported
- dual external reference mode supported
- new saving section in GUI
- time constant range massively increased
- added section on operation modes
- multi-frequency kit and real-time kit are different for HF2LI and HF2IS
- updated product selector
- added ultra-high stability option
- updated all images of graphical user interface

Revision 2002, 5-Aug-2009:

- first version of HF2 Series user manual

Table of Contents

Declaration of Conformity VIII
1. Getting Started 9
    1.1. Inspect the Package Contents 10
    1.2. Software Installation 12
2. Functional Overview 24
    2.1. Features 25
    2.2. Front Panel Tour 28
    2.3. Back Panel Tour 30
    2.4. Ordering Guide 31
    2.5. Operating Modes 33
3. Tutorials 38
    3.1. Tutorial HF2LI First Time User 39
    3.2. Tutorial Simple Loop 49
    3.3. Tutorial Dynamic Signals 55
    3.4. Tutorial External Reference 60
    3.5. Tutorial Noise Measurement 66
    3.6. Tutorial Amplitude Modulation 67
    3.7. Tutorial Frequency Modulation 72
    3.8. Tutorial Phase Locked Loop 79
    3.9. Tutorial PLL/Resonator 84
    3.10. Tutorial PID Controller with Auto Tune 87
    3.11. Tutorial PID Controller Tuning Tools 91
4. Functional Description HF2LI 101
    4.1. Graphical User Interface Overview 102
    4.2. Settings Tabs 104
    4.3. Other Settings 129
    4.4. Tools Tabs 139
5. Functional Description HF2IS 162
    5.1. Graphical User Interface Overview 163
    5.2. Settings Tabs 165
    5.3. Other Settings 170
    5.4. Tools Tabs 180
6. Communication and Connectivity 203
    6.1. Instrument Connectivity Overview 204
    6.2. ziServer's Text-based Interface 209
    6.3. Connecting to ziServer over insecure or firewalled networks 218
7. Node Definitions 221
    7.1. Overview 222
    7.2. Nodes 229
8. Real-time Option 282
    8.1. Installation of the Real-time Development Environment 283
    8.2. Real-Time Option Reference Manual 287
9. Specifications 305
    9.1. General Specifications 306
    9.2. Analog Interface Specifications 308
    9.3. Digital Interface Specifications 312
    9.4. Performance Diagrams 315
    9.5. Ground and Earth Scheme 325
    9.6. Reference Images 327
    9.7. Test Specifications 331
10. Signal Processing Basics 349
    10.1. Principles of Lock-in Detection 350
    10.2. Signal Bandwidth 353
    10.3. Discrete Filter 355

10.4. Full Range Sensitivity 357
10.5. Sinc Filtering 359
10.6. Zoom FFT 362
11. HF2CA Current Amplifier Datasheet 364
11.1. Key Features 365
11.2. Specifications 366
11.3. Functional Description 369
11.4. Applications 371
11.5. Cable Recommendation 375
12. HF2TA Current Amplifier Datasheet 376
12.1. Key Features 377
12.2. Specifications 378
12.3. Functional Description 381
12.4. Applications 383
12.5. Performance Tests 389
12.6. Cable Recommendation 393
Glossary 394
Index 400

Declaration of Conformity

The manufacturer

Zurich Instruments
Technoparkstrasse 1
8005 Zurich
Switzerland

declares that the product

HF2 Series (HF2LI, HF2IS), 50 MHz, 210 MSamples/s

fulfils the requirements of the European guidelines

2004/108/EC Electromagnetic Compatibility
2006/95/EC Low Voltage

The assessment was performed using the directives according to Table 1 .

Table 1. Conformity table

EN 61326-1:2006	Emissions for industrial environments, immunity for industrial environments
EN 55011	Group 1, class A and B (the product was tested in typical configuration)
EN 61000-4-2	CD 4 kV, AD 8 kV
EN 61000-4-3	10 V/m 80% AM 80 MHz - 1 GHz
	3 V/m 80% AM 1 MHz - 2 GHz
	1 V/m 80% AM 2 MHz - 2.7 GHz
EN 61000-4-4	2 kV power line
	1 kV USB line
EN 61000-4-5	1 kV line-line, 2 kV line-earth
EN 61000-4-6	3 V 80% AM, power line
EN 61010-1:2001	Safety requirements for electrical equipment for measurement, control and laboratory use

Figure 1. CE Logo

Chapter 1. Getting Started

Welcome to the world of the Zurich Instruments HF2. This first chapter supports you the intitial set-up of your HF2 Instrument and prepares for your first measurements. You are going to be assisted through:

- Inspecting the package content and accessories
- Installing the HF2 Instrument software on your host computer
- Powering-on the device, connecting the device via USB, and performing basic operation checks on the instrument
- List of essential handling and safety instructions
- Handy list of troubleshooting help

This chapter is delivered as hard copy with all initial instrument delivery to customers. It is also integral part of the HF2 User Manual.

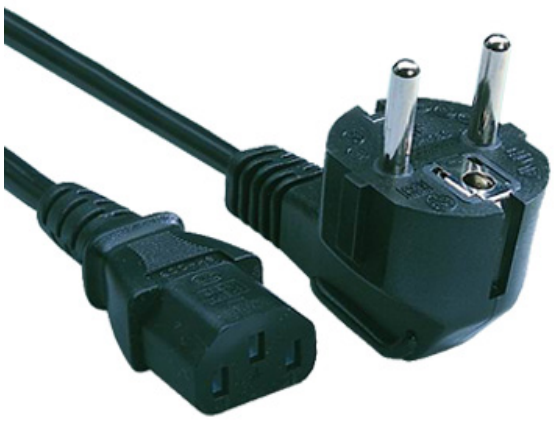
1.1. Inspect the Package Contents

If the shipping container appears to be damaged, keep the container until you have inspected the contents of the shipment and have performed basic functional tests.

You must verify that:

- You have received 1 Zurich Instruments HF2 Instrument
- You have received 1 power cord with a power plug suited to your country
- You have received 1 USB cable
- A printed version of the "Getting Started" section
- Additional cables have been added to the delivery if an HF2 pre-amplifier has been delivered at the same time
- The line voltage selector on the HF2 Instrument power inlet indicates the correct line voltage of your country (115V/60 Hz, or 230V/50Hz). While Zurich Instruments configures the power system when an instrument is initially delivered, no liability derives from potential wrong configuration or incorrect configuration at any point in time during the lifetime of the instrument
- The "Next Calibration" sticker on the rear panel of the Instrument indicates approximately 2 years ahead in time. Zurich Instruments recommends calibration intervals of 2 years
- For Japanese users only: you are supposed to operate the HF2 Instruments with an external 100 V to 110 V transformer in order to have reliable measurement results. Please verify having received the transformer included in your delivery from the local distributor.

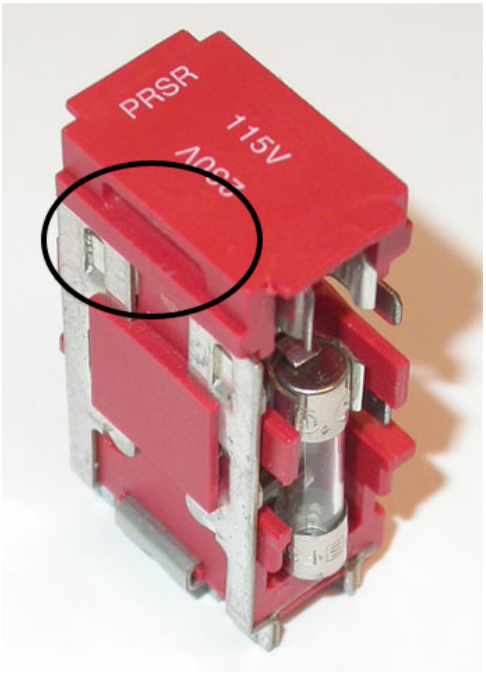
Table 1.1. Package contents for HF2 Instruments



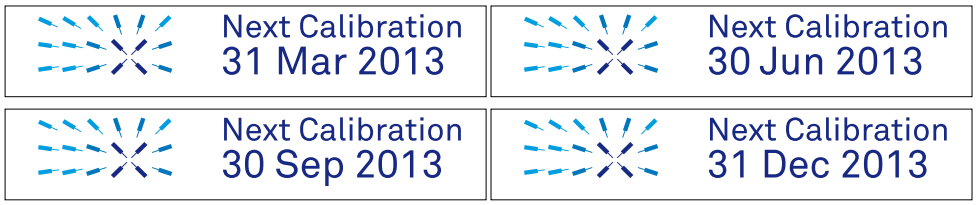
the USB cable

the power cord

the power inlet with selected 230V/50Hz power system



the fuse holder - requires 2 x 20 mm fast-acting fuses with 0.4 current limit - to extract the fuse holder use a small screwdriver in the indicated spot to lift it out of the casing



the "Next Calibration" sticker on the rear panel of your instrument

for Japanese users only: the 100 V to 110 V transformer

Carefully inspect your HF2 Instrument. If there is mechanical damage or the instrument does not seem to operate after the software installation , please consult the handling instructions and the troubleshooting , then notify the Zurich Instruments support team at <support@zhinst.com> as soon as possible.

1.2. Software Installation

HF2 Instruments are not stand-alone devices and as such the PC from which they are controlled must have the following two software packages installed:

1. The LabOne package, which installs the Data Server program, documentation and the Matlab, C and LabVIEW APIs.
2. The ziControl package, which installs the graphical user interface for controlling HF2 Instruments.

Instructions for downloading the correct version of the software packages from the Zurich Instruments website are described below.

1.2.1. Windows Installation

Requirements

Please ensure that the following requirements are fulfilled before trying to install the LabOne and ziControl software packages:

1. Your PC is running a version of Windows supported by Zurich Instruments. Currently both 32-bit and 64-bit architectures are supported on:
   - Windows XP,
   - Windows Vista,
   - Windows 7, and
   - Windows 8.
2. You have Windows administrator rights on the target system. Note after installation administrator rights are not required to run the Zurich Instruments software.
3. The .NET Framework installed on the target installation PC, see the section called "Windows .NET Framework Requirement" .
4. The correct version of the LabOne and ziControl installation packages for your operating system and platform have been downloaded from the Zurich Instruments downloads page [http://www.zhinst.com/downloads] (login required):
   - LabOne<platform>-<release>.<revision>.msi, for example:

     ```
     LabOne64-14.08.26000.msi
     ```

   - ziControl-<release>.<revision>-<platform>.zip, for example:

     ```
     ziControl-14.08.26000-win.zip
     ```

   Please ensure you download the correct architecture (32-bit/64-bit) of the LabOne installer. In order to determine which architecture is required see the the section called "Determining the PC Architecture on Microsoft Windows" . Note, there is only one version of the ziControl installer for use on both 32-bit and 64-bit systems.

After ensuring that the requirements above are fulfilled, please proceed to first install the LabOne package as described in the section called "Windows LabOne Installation" and then the ziControl package as described in the section called "Windows ziControl Installation" .

Note

In the naming convention the <platform> refers to the Windows addressing mode (32-bit: win32, 64-bit: win64), <release> refers the to software release as published by Zurich Instruments, and <revision> refers to the revision tag (which as no particular meaning for the user).

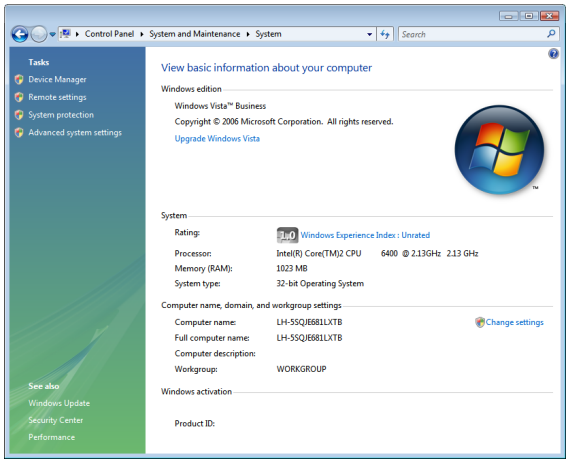
Determining the PC Architecture on Microsoft Windows

You can determine which architecture you are using as follows:

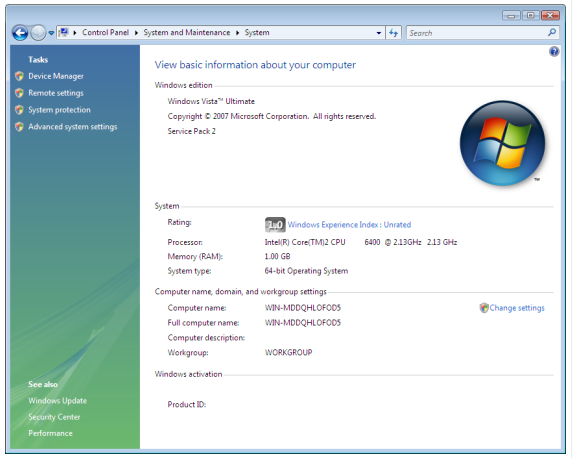
- Windows Vista: Control Panel → System and Maintenance → System / System type
- Windows 7: Control panel → System and Security → System / System type
- Windows 8: Control panel → System → System / System type

In general, x86 indicates a 32-bit system and x64 indicates a 64-bit operating system.

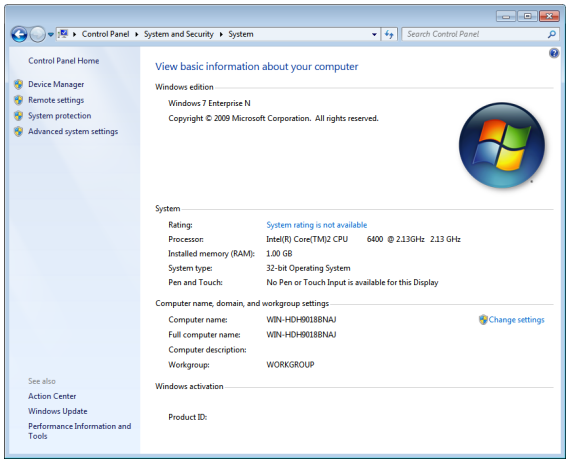
Table 1.2. Determining the PC's platform architecture (32-bit/64-bit).



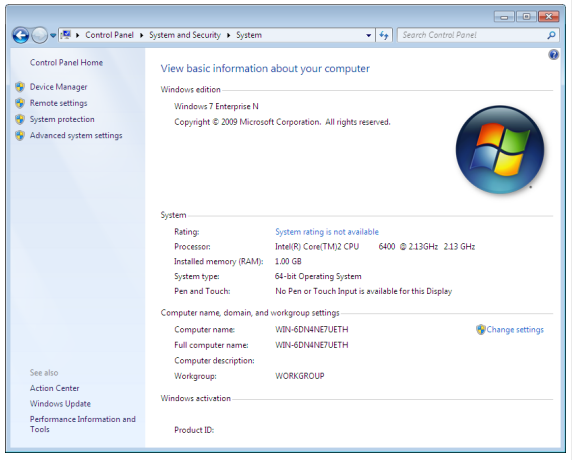
Windows Vista (32-bit: x86)



Windows Vista (64-bit: x64)



Windows 7 (32-bit: x86)



Windows 7 (64-bit: x64)

Windows .NET Framework Requirement

Zurich Instruments software requires the Microsoft .NET Framework to be installed on the host computer. The installation of LabOne will fail if this is not the case. It is possible to check whether the Microsoft .NET Framework is installed on your system under Windows Start -> Control panel -> Add and Remove Programs.

The minimum requirement is Microsoft .NET Framework 3.5 Service Pack 1. In case the required version is not installed, it can be installed through Windows Update tool (Windows Start -> Control panel -> Windows Update).

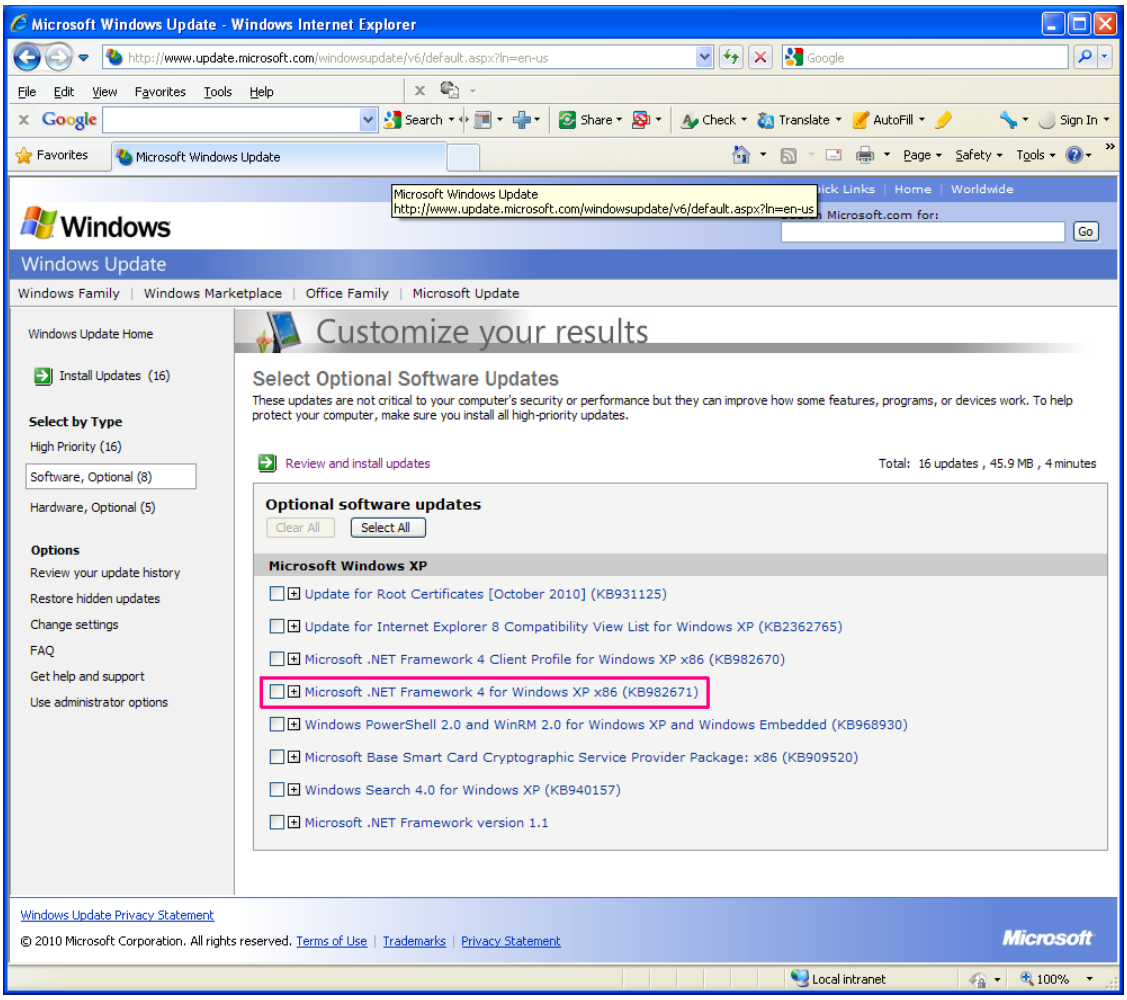

Microsoft Windows Update - Windows Internet Explorer
http://www.update.microsoft.com/windowsupdate/v6/default.aspx?ln=en-us
Windows Update
Windows Family | Windows Marketplace | Office Family | Microsoft Update
Windows Update Home
Install Updates (16)
Select by Type
High Priority (16)
Software, Optional (8)
Hardware, Optional (5)
Options
Review your update history
Restore hidden updates
Change settings
FAQ
Get help and support
Use administrator options
Customize your results
Select Optional Software Updates
These updates are not critical to your computer's security or performance but they can improve how some features, programs, or devices work. To help protect your computer, make sure you install all high-priority updates.
Review and install updates
Total: 16 updates , 45.9 MB , 4 minutes
Optional software updates
Clear All
Select All
Microsoft Windows XP
Update for Root Certificates [October 2010] (KB931125)
Update for Internet Explorer 8 Compatibility View List for Windows XP (KB2362765)
Microsoft .NET Framework 4 Client Profile for Windows XP x86 (KB982670)
Microsoft .NET Framework 4 for Windows XP x86 (KB982671)
Windows PowerShell 2.0 and WinRM 2.0 for Windows XP and Windows Embedded (KB968930)
Microsoft Base Smart Card Cryptographic Service Provider Package: x86 (KB909520)
Windows Search 4.0 for Windows XP (KB940157)
Microsoft .NET Framework version 1.1
Windows Update Privacy Statement
© 2010 Microsoft Corporation. All rights reserved. Terms of Use | Trademarks | Privacy Statement
Microsoft

Figure 1.1. Installation of the .NET Framework.

Windows LabOne Installation

1. The HF2 Instrument should not be connected to your computer during the installation process
2. Start the `msi` windows-installer program and follow the instructions. Please note that your windows user account needs to have administrator rights for this action. The installation proceeds as follows:
   - On the welcome screen click the "Next" button

- After reading through the Zurich Instruments end-user license agreement, check the "I accept the terms in the License Agreement" check box and click the "Next" button
- Review the features you want to have installed and then click the "Next" button - it is recommended to install the complete package
- Click the "Install" button to start the installation process
- Accept the installation of new software (up to 3 pop-up windows) on your computer including the installation of USB device drivers - you may decide to select the "Always trust software from Zurich Instruments AG" switch
- A message box appears reminding to complete the installation of the software by connecting the HF2 Instrument to the computer by means of the USB cable - click the "OK" button

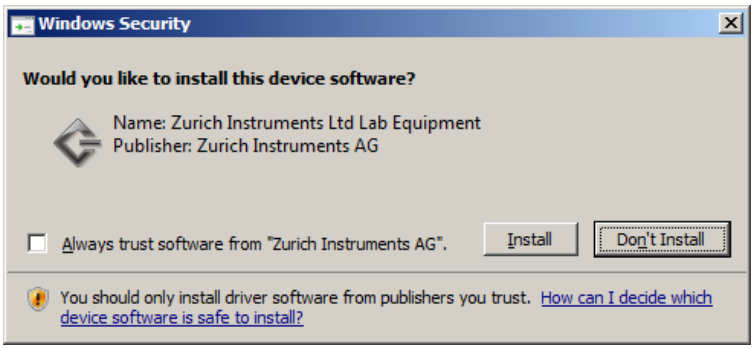

Windows Security
Would you like to install this device software?
Name: Zurich Instruments Ltd Lab Equipment
Publisher: Zurich Instruments AG
Always trust software from "Zurich Instruments AG".
Install
Don't Install
You should only install driver software from publishers you trust. How can I decide which device software is safe to install?

Figure 1.2. Pop-up Window 1: Confirm to install the USB drivers (up to two times).

3. Click "Finish" to close the installer

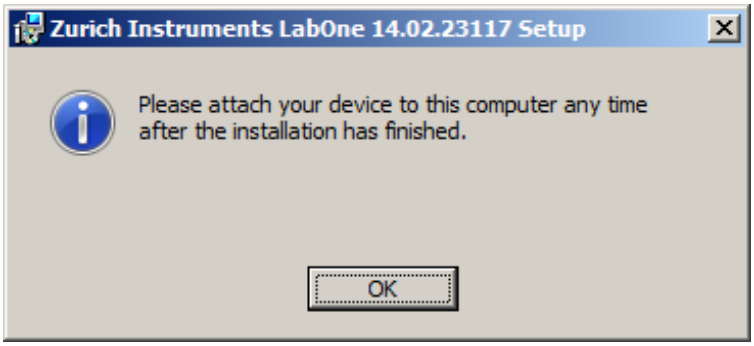

Zurich Instruments LabOne 14.02.23117 Setup
Please attach your device to this computer any time after the installation has finished.
OK

Figure 1.3. Pop-up Window 2: Click 'Ok' to finish the installation.

4. Complete the LabOne installation process as follows:
   - The HF2 Instrument should not be connected to your computer during the installation process
   - Connect the power cord to the power input on the rear panel of the device and to a suited AC power outlet
   - Turn on the HF2 Instrument by means of the power switch on the back panel. Check if the blue power indicator LED on the front panel is lit
   - Connect the HF2 to the host computer with the provided USB cable. You need to have administrator rights to finish the installation of the software. You may have to click through the common installation screens on older versions of Windows. In this step the installation of the USB driver completes

When the installation is completed, and every time you re-connect the instrument to your computer, the red "Over" LEDs on the HF2 front panel will blink 10 times to indicate that a

functional computer connection is established. The HF2 Instrument is now powered on, and ready to be used.

With the installation of the LabOne package, you will be able to access the documentation: Windows → Zurich Instruments → Documentation → User Manual HF2.

Windows ziControl Installation

1. Start the `msi` windows installer
2. Choose the destination directory for the ziControl software, by default:

   ```
   C:\Program Files\Zurich Instruments\ziControl
   ```

   and for the National Instruments products, by default:

   ```
   C:\Program Files\National Instruments
   ```

   Click on "Next".
3. License agreement for National Instruments Software. The installer package contains the LabVIEW Run-Time Engine by National Instruments, which is free of charge. This Run-Time Engine must be installed on any Windows system where you plan to run executables that were built with the LabVIEW Application Builder. If you accept the license click on "I accept the license agreement" and click on "Next".
4. Summary of the installation, click "Next" to proceed to start the installation.
5. Click "Finish" to finish installation. All the required software in now installed on the computer.

Note

If you encounter problems regarding the application font, go to the installation directory of ziControl and adapt the font settings in the configuration file `ziControl.ini`. Alternatively, you can also start the application `ziFontConfig.exe` (located in the installation directory) with elevated access rights in order to adjust it automatically.

64-bit Windows Installer Issue

In seldom cases the Windows Installer Service may be corrupted resulting in error message:

```
The Windows Installer Service could not be accessed. This can occur if the Windows Installer is not correctly installed. Contact your support personnel for assistance.
```

Assuming the Windows Installer itself is installed (which is the case on most Windows systems), this problem can be caused by the 32-bit service being started instead of the 64-bit version. To resolve this issue, the following steps have to be performed:

1. Start the `cmd` command prompt with administrator privileges.
2. Run the command `net stop MSIServer` in the `cmd` window to stop the service.
3. Run the `regedit` Windows Registry Editor.
4. Locate the key

`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\MSIServer`

and set the value of the entry `WOW64` to 0.

5. Close the Windows Registry Editor and execute the command `net start MSIServer` in the `cmd` window to restart the service. Confirmation of a successful service start should be printed and the installation of 64-bit software should be possible again.

1.2.2. Linux Installation

Requirements

Please ensure that the following requirements are fulfilled before trying to install the LabOne and ziControl software packages:

1. Officially, only Ubuntu 12.04 LTS (i386, amd64) is supported although in practice the LabOne and ziControl packages could work on other platforms. Please ensure that you are using a Linux distribution that is compatible with Ubuntu/Debian, but preferably Ubuntu 12.04 LTS.
2. You have administrator rights for the system.
3. The correct version of the LabOne and ziControl installation packages for your operating system and platform have been downloaded from the Zurich Instruments downloads page [http://www.zhinst.com/downloads] (login required):
   - LabOneLinux<arch>-<release>.<revision>.tar.gz, for example:

     `LabOneLinux64-14.08.26000.tar.gz`
   - ziControl-<release>.<revision>-linux.tar.gz, for example:

     `ziControl-14.08.26000-linux.tar.gz`

   Please ensure you download the correct architecture (32-bit/64-bit) of the LabOne installer. The `uname` command can be used in order to determine which architecture you are using, by running:

   ```
   uname -m
   ```

   in a command line terminal. If the command outputs "`x686`" the 32-bit version of the LabOne package is required, if it displays "`x86_64`" the 64-bit version is required. Note, there is only one version of the ziControl installer for use on both 32-bit and 64-bit systems.
4. This step is only necessary when using ziControl on 64-bit platforms. Ensure that the `ia32_libs` package is installed by typing:

   ```
   sudo apt-get install ia32-libs
   ```

   in a command-line windows. This is necessary to run the 32-bit LabVIEW Run-Time on 64-bit machines.

Linux LabOne Installation

Please proceed with the installation in a command line shell as follows:

1. Extract the LabOne tarball in a temporary directory:

```
tar xzvf LabOneLinux<arch>-<release>-<revision>.tar.gz
```

2. Navigate into the extracted directory.

```
cd LabOneLinux<arch>-<release>-<revision>
```

3. Run the install script with administrator rights and proceed through the guided installation, using the default installation path if possible:

```
sudo bash install.sh
```

The install script lets you choose between the following three modes:

- Type "a" to install the Data Server program, documentation and APIs.
- Type "u" to install the `udev` support.
- Type "ENTER" to install both the options "a" and "l" (recommended).

4. Optional: If `udev` was not installed, the Data Server program for the HF2 "ziServer" should be started manually:

```
ziServer &
```

5. Optional: If `udev` was installed, then the Data Server program does not have to be started manually, it will be started upon plugging in the USB cable of the HF2 and powering on the device. In this case the ziService program will also have been installed which can be used to control ziServer (with administrator rights).

```
sudo ziService [start,stop,update,status]
```

Important

Do not use two ziServer instances running in parallel. If you have `udev` support installed, ziServer is started the first time you are powering on a HF2 Instrument (hot plug). It is not necessary to manually start the program.

Linux ziControl Installation

Please ensure that the LabOne package has been installed as described above in the section called “Linux LabOne Installation”. The ziControl installer package optionally installs the LabVIEW Run-Time Engine by National Instruments, please install it to use ziControl if it's not already available on your system.

Please proceed with the installation in a command line shell as follows:

1. Extract the ziControl tarball in a temporary directory:

```
tar xzvf ziControl-<release>-<revision>-linux.tar.gz
```

2. Navigate into the extracted directory.

```
cd ziControl-<release>-<revision>-linux
```

3. Run the install script with administrator rights and proceed through the guided installation, using the default installation path if possible:

```
sudo bash install.sh
```

The install script lets you choose between the following three modes:

- Type "a" to install ziControl and ziFontConfig.

- Type "l" to install the LabVIEW Run-Time Engine 2010 SP1 32-bit from National Instruments as a deb-package.
- Type "ENTER" to install both the options "a" and "l".

4. Start the graphical user interface ziControl by typing:

```
ziControl &
```

Note

If you encounter problems regarding the application font, go to the installation directory of ziControl and adapt the font settings in the configuration file `ziControl.ini`. Alternatively, you can also invoke the shell command `gksu ziFontConfig` in order to adjust it automatically.

Uninstalling the LabOne and ziControl packages on Linux

Both software packages copy an uninstall script to the base installation paths (the default installation directory is `/opt/zi/`). To uninstall the packages please proceed in a command line shell as follows:

1. Navigate to the base installation path of ziControl and LabOne, if the default path is used:

```
cd /opt/zi/
```

2. Run the uninstall scripts with administrator rights and proceed through the guided steps:

```
sudo bash uninstall_LabOne<arch>-<release>-<revison>.sh
```

```
sudo bash uninstall_ziControl-<release>-<revison>.sh
```

1.2.3. Handling and Safety Instructions

The HF2 Instrument is an electronic device. It is seriously discouraged to open the device, as there are high-voltage parts inside which may be harmful to human beings. There are no serviceable parts inside the device. Opening the device immediately cancels the warranty as provided by Zurich Instruments.

The following general safety instructions must be observed during all phases of operation, service, and handling of the instrument. The disregard of these precautions and all specific warnings elsewhere in this manual may affect correct operation of the equipment and its lifetime.

Zurich Instruments assumes no liability for the user's failure to observe and comply with the instructions in this user manual.

Table 1.3. Safety Instructions

Ground the instrument	The chassis must be correctly connected to earth ground by means of the supplied power cable. Alternatively also the ground pin on the rear panel can be used. This avoids electrical shocks and potential damage to the instrument. For more information on ground and earth scheme, refer to section Section 9.5

Maximum ratings	The specified electrical ratings for the connectors of the instrument should not be exceeded at any time during operation (please refer to section Section 9.1)
Do not service or adjust anything yourself	There are no serviceable parts inside
Software updates	Regular software updates provide the user with many important bug fixes, but also with many new features. Only the last released software version will be supported by Zurich Instruments
Overseas travel	Consider that a power system change without changing the orientation of the fuse holder will damage the fuses, or make the instrument behaving unpredictably
Warnings	Instructions contained in any warning issued by the instrument, either by the software, the graphical user interface, or mentioned in this manual must be followed
Notes	Instructions contained in the notes of this user manual are of essential importance for the correct interpretation of the acquired measurement data
Location and ventilation	Keep and operate the HF2 Instrument in a dry location that suits the general specifications. Do not block the ventilator opening on the back or the air intake on the side of the chassis and allow a reasonable space for the air to flow
RJ45 plugs	Although the device has several RJ45 plugs, these are not intended for Ethernet connection. Connecting these plugs with an Ethernet device may damage the HF2 Instrument and/or the Ethernet device
Operation and storage	Do not operate or store at a location outside the specified ambient conditions (please refer to section Section 9.3)
Handling	Do not throw the HF2 Instrument, handle with due care, do not store liquids on the device as there is a chance of spilling and damage

When you notice any of the situations listed below, immediately stop the operation of the HF2 Instrument, disconnect the power cord, and contact the support team at Zurich Instruments, either through the website form or by email at <support@zhinst.com> .

Table 1.4. Unusual Conditions

Fan is not working properly or not at all	Operation must be avoided to prevent overheating of sensitive electronic components
Power cord or power plug on instrument is damaged	Operation must be avoided in order to prevent overheating, electric shock, or fire. Please exchange the power cord with a quality product
Instrument emits abnormal noise, smell, or sparks	Operation must be avoided to prevent large damage

1.2.4. Troubleshooting

Your HF2 Instrument is an advanced equipment that has many more settings, buttons and switches compared to traditional lock-in amplifiers. Whereas the users will profit from the features and flexibility, the control panels might appear overwhelming to the first time user, and even expert users might be surprised by strange combinations of settings. The instrument has been tested to perform according to specifications, and therefore regular checking of the settings will prevent wrong measurements. It is also advisable to use the instrument base on known configurations as it is possible to store all settings in specific external files.

This section contains provides an easy to follow checklist and specific solution to many typical issues. It is also advised to regularly keep this list in mind to avoid wrongly acquired measurement data.

The software cannot be installed or uninstalled 1: please verify that you have Windows administrator rights. If you receive a message saying that the Windows Installer Service can not be accessed, please see the section called “64-bit Windows Installer Issue” .

The software cannot be installed or uninstalled 2: please verify that the computer being used for software installation complies to the requirements mentioned in Table 9.3 .

The HF2 ziControl starts with unexpected messages: although Windows XP is fully supported by the HF2 software, Windows XP is known for its USB weaknesses that can generate conflicts between drivers and/or some USB peripherals (e.g. web cams). On Windows XP even non-aggressor devices can lead to conflicts. This problem can also occur with later versions of Windows, but less likely. For computers with many USB peripherals it might be beneficial to reduce the unnecessary ones and to update the drivers.

The HF2 Instrument does not turn on: please verify the power supply connection, the power-on switch on the back panel of the instrument, and the fuse holder. For verifying the red fuse holder, check whether the fuse is blown (e.g. sign of burn mark and burned wire inside the glass tube) or if it has not popped out of the red holder. The fuse must have only one end clamped between two metal plates inside the red holder and not both ends. The other end of the fuse should be touching a tilted horizontal metal plate.

The HF2 Instrument turns on but delivers obviously wrong measurements: please verify the power system setting on the back panel of the device is set to the power system of your country (110 V / 60 Hz, 220 V / 50 Hz). Make sure the fuse holder is set to the correct power supply position. This means that the wanted power supply label, 230 V or 115 V, must be positioned beside the edge of the power socket (e.g. not beside the power switch).

The HF2 Instrument performs poorly in a country with 100 V supply system (e.g. Japan): if no 100 V to 110 V transformer is used, the internal power supplies might be below specifications and some circuits might perform worse than specification. Users in countries with 100 V supply system are warmly recommended to use an external transformer (delivered with the instrument).

The HF2 Instrument shows limited data throughput on USB: although the host computer requirements are not particularly demanding, highest performance in USB throughput will require a performing desktop. The USB might be limiting the data throughput w.r.t. specification Table 9.4 . Many concurrent transfers on the USB will limit the individual transfer. In particular the oscilloscope should be turned-off when not needed by the application. Th status of the USB transfer can be continuously monitored in the status tab.

The HF2 Instrument performs poorly in single-ended operation: the signal inputs of the instrument might be set to differential operation. Please verify to turn-off the differential input (switch on the graphical user interface). Alternatively use male grounding caps on the negative input.

The HF2 Instrument has a high input noise floor: There may be crosstalk through the USB ground from the computer into the HF2 Instrument. In order to decouple the computer from the Instrument a USB electrical isolator could be employed, e.g. the USB 2.0 Ranger 2201 from Icron Technologies used with a UTP Cat 5 or 6 cable (UTP stands for "unshielded twisted pair"). Further the switched power supply of the ranger may need to be exchanged with another 5 V source. In-house testing has been performed with the now obsoleted USB Ranger 2101. The ground connection between the computer and the Instrument will be high impedance with this solution.

The HF2 Instrument performs poorly at low frequencies (below 10 kHz): the signal inputs of the instrument might be set to AC operation. Please verify to turn-off the AC switch on the user interface.

The HF2 Instrument performs poorly during operation 1: the demodulator filters might be set too wide (too much noise) or too narrow (not enough signal) for your application. Please verify if the demodulator filter settings match your frequency versus noise plan.

The HF2 Instrument performs poorly during operation 2: clipping of the input signal may be occurring. This is detectable by monitoring the red LEDs on the front panels or on the status tab on the graphical user interface. This can be avoided by adding enough margin on the input range setting (for instance 50% to 70% of the maximum signal peak.

The HF2 Instrument performs strangely when working with the HF2LI-MF and HF2IS-MF options: it is easy to turn-on more signal generators than initially planned. Check the generated Signal Output with the integrated oscilloscope and check the number of simultaneously activated oscillator voltages.

The HF2 Instrument performs close to specification, but higher performance is expected: after 2 years since the last calibration, a few analog parameters are subject to drift. This may provoke inaccurate measurements. Zurich Instruments recommends to re-calibrate the HF2 Instrument every 2 years.

The HF2 Instrument measurements are unpredictable: please monitor the status tabs if any of the warning is occurring or has occurred in the past.

The HF2 Instrument does not generate any output signal: verify that signal output switch has been activated the related control panel.

The HF2 Instrument locks poorly using the digital I/O as reference: make sure that the digital input signal has a high slew rate and clean level crossings.

The HF2 Instrument locks poorly using the auxiliary analog inputs as reference: the input signal amplitude might be too small. Use proper gain setting of the input channel.

Location of the log files

One will find the log files in the following directories on Windows 8, Windows 7 and Windows Vista:

- LabOne Data Server: `C:\Users\[USER]\AppData\Local\Temp\Zurich Instruments\LabOne\ziDataServerLog`
- LabOne Web Server: `C:\Users\[USER]\AppData\Local\Temp\Zurich Instruments\LabOne\ziWebServerLog`
- ziServer (HF2 Instrument)
  - started by service: `C:\Windows\Temp\ziServerLog`
  - started manually: `C:\Users\[USER]\AppData\Local\Temp\ziServerLog`
- ziControl (HF2 Instrument): `C:\Users\[USER]\Documents\LabVIEW Data` and the file name is called `com.zhinst.ziControlStatusLog.txt`.

On Windows XP:

- LabOne Data Server UHF : `C:\Documents and Settings\[USER]\Local Settings\Temp\Zurich Instruments\LabOne\ziDataServerLog`
- LabOne Web Server UHF: `C:\Documents and Settings\[USER]\Local Settings\Temp\Zurich Instruments\LabOne\ziWebServerLog`
- ziServer (HF2 Instrument)
  - started by service: `C:\WINDOWS\Temp\ziServerLog`
  - started manually: `C:\Documents and Settings\[USER]\Local Settings\Temp\ziServerLog`
- ziControl (HF2 Instrument): `C:\Documents and Settings\[USER]\Documents\LabVIEW Data` and the file name is called `com.zhinst.ziControlStatusLog.txt`.

Chapter 2. Functional Overview

This chapter helps you to quickly get acquainted with the main features, the panels, and the operating modes of the HF2 Series. A product selector is provided listing the key features of the products in order to support the selection and ordering. This section is intended as overview and therefore has a coarse level of detail without containing detailed descriptions.

2.1. Features

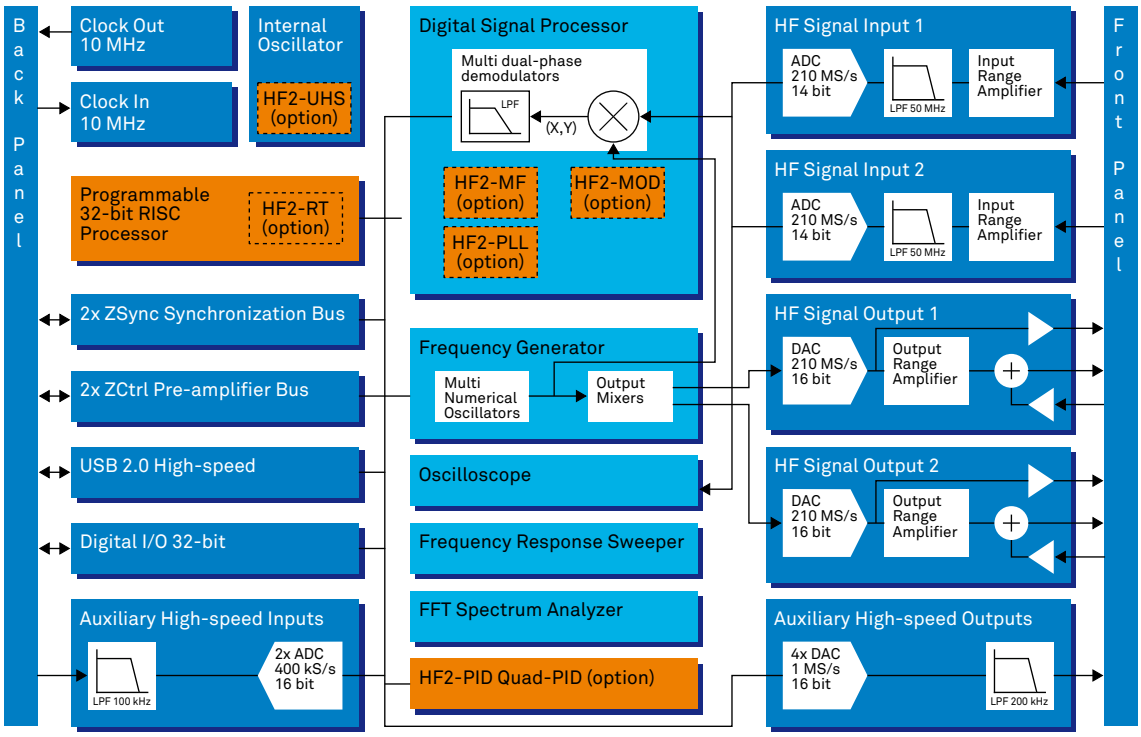


Figure 2.1. HF2 functional diagram

The HF2 Instrument as in Figure 2.1 consists of 4 high-frequency analog blocks, 2 low-frequency auxiliary blocks, the internal digital processing block (light-blue), and the hardware interfaces (mostly available on the back panel of the instrument).

The signal to be measured is connected to one of the two high-frequency analog inputs where it is amplified to a defined range, filtered, and digitized at very high speed. The resulting samples are fed into the digital signal processing block for demodulation by means of up to 8 dual-phase demodulators. The demodulators output samples flow into the embedded RISC processor for further processing or to be sent to the host computer. The samples are also sent to the auxiliary outputs in order to be available on the front panel of the HF2 Instrument.

The numerical oscillators generate sine and cosine signal pairs that are used for the demodulation of the input samples and also for the generation of the high-frequency output signals. For this purpose, the Output Mixers generate a weighted sum of the generator outputs to generate the multi-frequency signal that can be used a stimulation signal. The 2 high-frequency output stages provide analog to digital conversion, signal scaling (range), add of an external AC or DC signal, and a synchronization signal.

Operating Modes

- Internal reference mode
- External reference mode
- Auto reference mode
- Dual-channel operation
- Dual-harmonic mode
- Multi-harmonic mode

- Arbitrary frequency mode

High-frequency Analog Inputs

- 2 low-noise high-frequency inputs
- Differential & single-ended operation (A, -B, A-B)
- Variable input range
- Variable input impedance
- AC/DC coupling

High-frequency Analog Outputs

- 2 low-noise high-frequency outputs
- Large output range
- Variable output range settings
- 1 synchronization signal for each output
- 1 adder signal for each output

Auxiliary Analog Input/Outputs

- 4 auxiliary high-speed outputs
- 2 auxiliary high-speed inputs
- User defined signal on auxiliary output

Demodulators & Reference

- Up to 8 dual-phase demodulators
- Up to 8 programmable numerical oscillators
- Programmable demodulators filters
- Very-high resolution internal reference
- 64-bit resolution demodulator outputs

Measurement Tools

- Spectroscope
- Numerical
- Oscilloscope
- Frequency response analyzer
- FFT spectrum analyzer

User-programmable Embedded Processor (Option)

- Microblaze 32-bit RISC
- 64 MHz operation allows implementation of real-time control loops
- 32-bit floating-point unit
- 64 kB internal memory (maximum program size)
- 64 MB external memory DDR2

Other Interfaces

- USB 2.0 high-speed 480 Mbit/s host interface

- DIO: 32-bit digital input-output port
- ZSync: 2 ports for inter-instrument synchronization bus (ZI proprietary)
- ZCtrl: 2 ports for control/power bus for external pre-amplifiers (ZI proprietary)
- Clock input connector (10 MHz)

Software Features

- ziControl, self-explanatory graphical interface with control of up 16 remote/local devices from one GUI
- ziServer multi-mode multi-connection server
- ziAPI for extended programmability in C, LabVIEW, MATLAB, and Python - programming examples included
- Console: text interface to connect virtually any programming language

2.2. Front Panel Tour

The front panel BNC connectors and control LEDs are arranged in 5 sections as shown in Figure 2.2 and Figure 2.3 listed in Table 2.1 . The HF2LI and HF2IS have the same connectors and connector functionality on their front- and back panel.

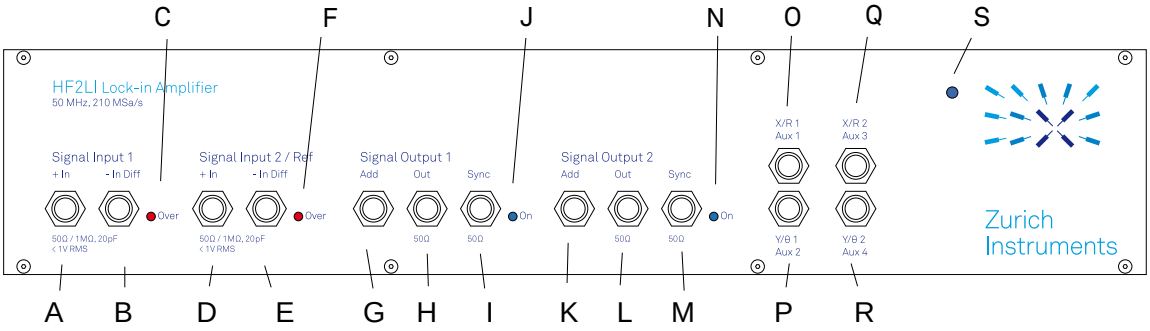


Figure 2.2. HF2LI front panel

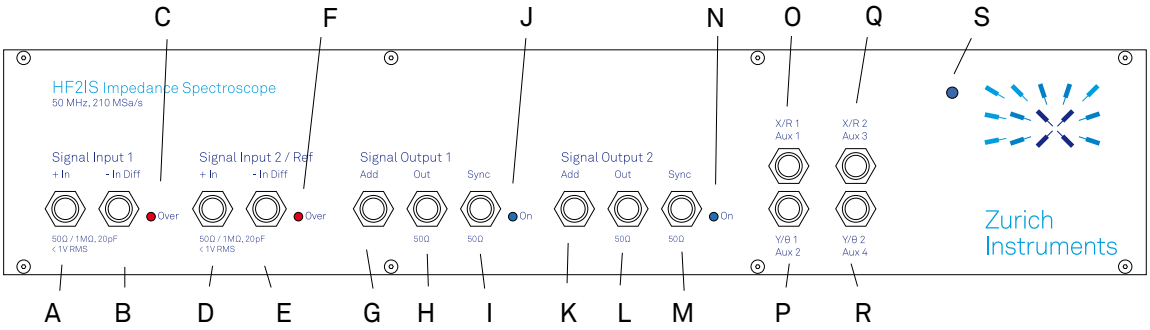


Figure 2.3. HF2IS front panel

Table 2.1. HF2 Series front panel description

Position	Label / Name	Description
A	Signal Input 1 + In	single-ended input
B	Signal Input 1 - In Diff	negative input (when not used, has to be internally shorted to ground with switch on graphical user interface)
C	Signal Input 1 Over	this LED indicates that the input signal saturates the A/D converter
D	Signal Input 2 / Ref + In	single ended input / reference input for external reference mode
E	Signal Input 2 / Ref - In Diff	negative input (when not used, has to be internally shorted to ground with switch on graphical user interface)
F	Signal Input 2 Over	this LED indicates that the input signal saturates the A/D converter
G	Signal Output 1 Add	the signal applied to the connector is added (analog add) to the output signal
H	Signal Output 1 Out	high-frequency output
I	Signal Output 1 Sync	the output signal before the output gain stage for use as synchronization or monitoring signal; the amplitude voltage

Position	Label / Name	Description
		calculates as ratio of the corresponding output amplitude and its range setting
J	Signal Output 1 On	this LED indicates that the signal output is turned on
K	Signal Output 2 Add	the signal applied to the connector is added (analog add) to the output signal
L	Signal Output 2 Out	high-frequency output
M	Signal Output 2 Sync	the output signal before the output gain stage for use as synchronization or monitoring signal; the amplitude voltage calculates as ratio of the corresponding output amplitude and its range setting
N	Signal Output 2 On	this LED indicates that the signal output is turned on
O	X/R 1 / Aux 1	this connector provides either the in-phase signal of the demodulator (X1), the magnitude (R1), or an auxiliary output signal Aux 1
P	Y/Θ 1 / Aux 2	this connector provides either the quadrature signal of the demodulator (Y1), the phase (Θ1), or an auxiliary output signal Aux 2
Q	X/R 2 / Aux 3	this connector provides either the in-phase signal of the demodulator (X2), the magnitude (R2), or an auxiliary output signal Aux 3
R	Y/Θ 2 / Aux 4	this connector provides either the quadrature signal of the demodulator (Y2), the phase (Θ2), or an auxiliary output signal Aux 4
S	Power	instrument mains power-on LED

2.3. Back Panel Tour

The back panel is the main interface for power, control, service and connectivity to other ZI instruments. Please refer to Figure 2.4 and Table 2.2 for the detailed description of the items.

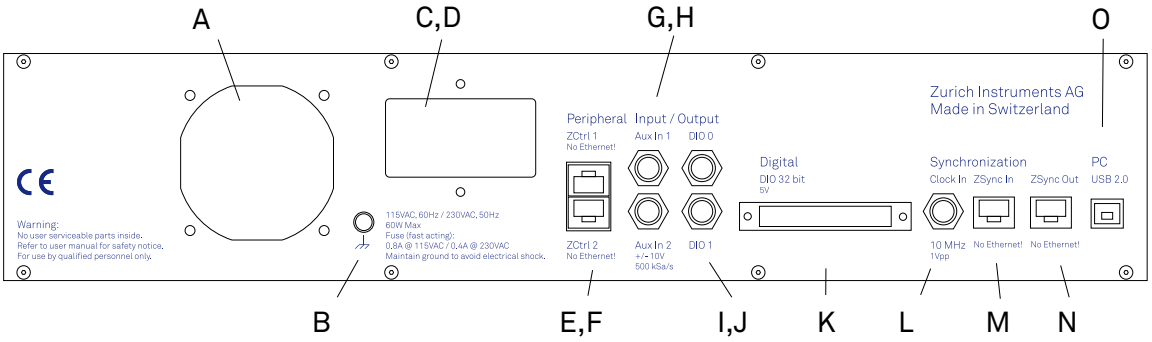


Figure 2.4. HF2 Series back panel

Table 2.2. HF2 Series back panel description

Position	Label / Name	Description
A	-	ventilator (important: keep clear from obstruction)
B	Earth ground	4 mm banana jack connector for earth ground purpose, electrically connected to the chassis and the earth pin of the power inlet
C	Power inlet	power inlet with On/Off switch
D	Power system	select between 115 V and 230 V power system
E	ZCtrl 1	peripheral pre-amplifier power & control bus 1 - attention: this is not an Ethernet plug, connection to an Ethernet network might damage the instrument
F	ZCtrl 2	peripheral pre-amplifier power & control bus 2 - attention: this is not an Ethernet plug, connection to an Ethernet network might damage the instrument
G	Aux In 1	auxiliary high-sampling rate input 1
H	Aux In 2	auxiliary high-sampling rate input 2
I	DIO 0	digital input/output 0
J	DIO 1	digital input/output 1
K	DIO	digital input/output 0-31
L	Clock In	clock input (10 MHz)
M	ZSync In	inter-instrument synchronization bus input - attention: this is not an Ethernet plug, connection to an Ethernet network might damage the instrument
N	ZSync Out	inter-instrument synchronization bus output - attention: this is not an Ethernet plug, connection to an Ethernet network might damage the instrument
O	USB	host computer connection

2.4. Ordering Guide

The HF2 Series is a product line comprising an impedance spectroscope and a digital lock-in amplifier covering advanced requirements for laboratory equipment. The HF2 Series provides best-in-class performance, wide operation range, intuitive handling and excellent accuracy.

Table 2.3 provides an overview of the available products in the HF2 Series. Upgradeable features are options that can be purchased anytime without need to send the instrument to Zurich Instruments - the upgradeable features consist of a firmware upgrade.

Table 2.3. HF2 Series product codes for ordering

Product code	Product name	Description	Upgrade in the field possible
HF2LI	HF2LI Lock-in Amplifier	base lock-in amplifier	-
HF2LI-MF	HF2LI-MF Multi-frequency	option	yes
HF2LI-RT	HF2LI-RT Real-time	option	yes
HF2LI-PLL	HF2LI-PLL Dual Phase-locked Loop	option	yes
HF2LI-PID	HF2LI-PID Quad PID Controller	option	yes
HF2LI-MOD	HF2LI-MOD AM/FM Modulation	option	yes
HF2LI-UHS	HF2LI-UHS Ultra-high Stability	option	no
HF2LI-WE2Y	HF2LI-WE2Y Warranty Extension 2 Years	warranty extension for 2 additional years	yes
-	-	-	-
HF2PLL	HF2PLL Phase-locked Loop	bundle of the HF2LI plus the HF2LI-PLL and the HF2LI-PID options	-
-	-	-	-
HF2IS	HF2IS Impedance Spectroscope	base impedance spectroscope	-
HF2IS-MF	HF2IS-MF Multi-frequency	option	yes
HF2IS-RT	HF2IS-RT Real-time	option	yes
HF2IS-UHS	HF2IS-UHS Ultra-high Stability	option	no
HF2IS-WE2Y	HF2IS-WE2Y Warranty Extension 2 Years	warranty extension for 2 additional years	yes
-	-	-	-
HF2CA	HF2CA Current Amplifier	low noise current amplifier with adaptable input impedance	yes
HF2TA	HF2TA Current Amplifier	low noise transimpedance amplifier	yes

Table 2.4. Product selector

Feature	HF2LI	HF2LI + HF2LI-MF	HF2IS	HF2IS + HF2IS-MF
Internal reference mode	yes	yes	yes	yes
External reference mode	yes	yes	-	-
Auto reference mode	yes	yes	-	-
Dual-channel operation (2 independent measurement units)	yes	yes	yes	yes
Sinusoidal generators	2	2	2	2
Superposed output sinusoidals per generator	1	up to 6	up to 4	up to 8
Dual-harmonic mode	yes	yes	-	-
Multi-harmonic mode	-	yes	-	-
Arbitrary frequency mode	-	yes	yes	yes
Number of demodulators	6	6	4	8
Simultaneous freq. supported (fundamentals/harmonics)	2/4	6/-	4/-	8/-
Signal input select switch matrix	-	yes	yes	yes
Oscillator select switch matrix	-	yes	-	-
50 MHz, 210 MS/s, 0.8µs TC	yes	yes	yes	yes
DSP technology	128 bit	128 bit	128 bit	128 bit
Dynamic reserve	120 dB	120 dB	-	-
Lock-in range	50 MHz	50 MHz	-	-
USB 2.0 480 Mbit/s	yes	yes	yes	yes
ziControl, ziAPI, ziServer software	yes	yes	yes	yes
Frequency response sweeper	yes	yes	yes	yes
Oscilloscope	yes	yes	yes	yes

2.5. Operating Modes

2.5.1. Internal Reference Mode

The internal reference mode takes advantage of the internal HF generators inside the HF2 Instrument. There are 6 frequency generators in the HF2LI and up to 8 frequency generators in the HF2IS. The output of these generators are added numerically inside the instrument avoiding complicated external analog signal adders and the resulting signal is fed to the device under test. The internal reference mode is the preferred mode as the signal recovery works at its best as the generated frequency is known inside of the instrument. The signal acquisition works immediately and there is no delay lock-time.

The internal reference mode is supported with single-channel and dual-channel operation. This is possible as the HF2 Instrument includes 2 independent measurement units that are working autonomously. Each of the measurement units provides analysis for one fundamental and 2 harmonic frequencies in parallel (sometimes called dual-harmonic mode). In total, the HF2LI can measure 2 fundamental and 4 harmonic frequencies, while the HF2IS can measure 4 frequencies. The number of frequencies increases with the multi-frequency options.

The demodulator samples are available in analog format on the auxiliary outputs of the HF2 Instrument and digitally on the connected computer transferred over the USB interface. The auxiliary outputs generate an analog signal after a linear digital to analog conversion at high sample rate. There are 2 pairs of analog signals allowing to output any 2 of the demodulation sample streams. All demodulator streams are available on the computer and can be further analyzed or stored in the local drives.

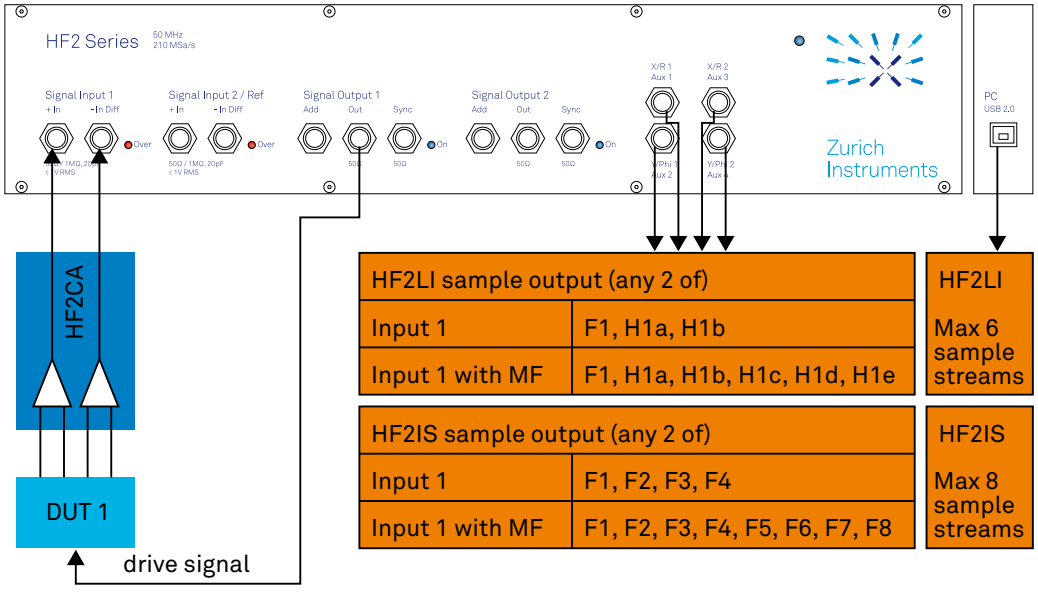


Figure 2.5. HF2 internal reference mode / single-channel

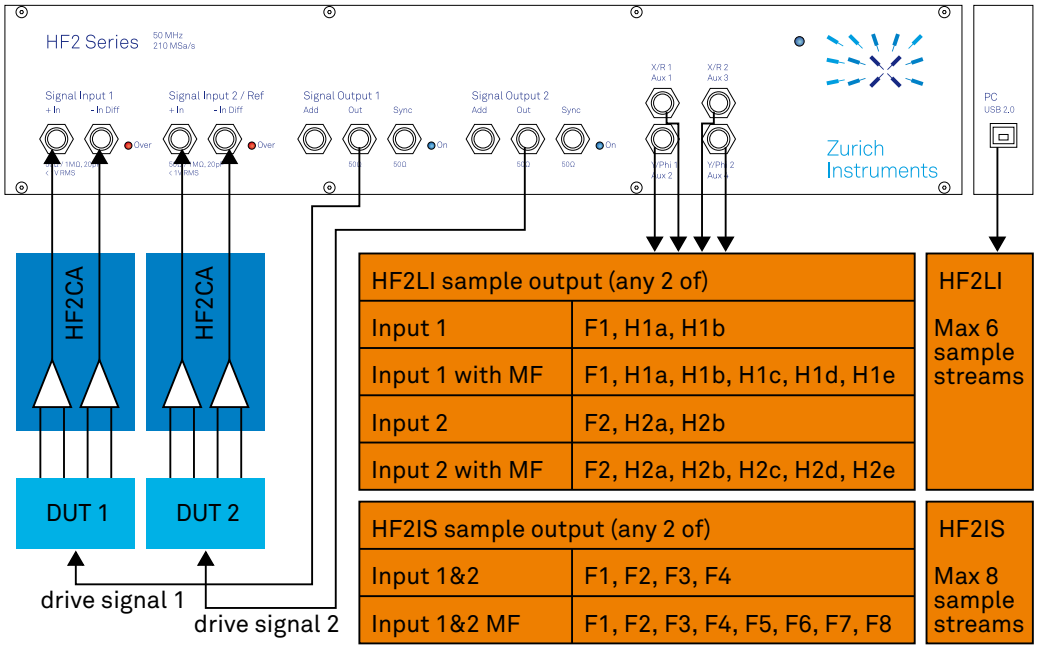


Figure 2.6. HF2 internal reference mode / dual-channel

2.5.2. External Reference Mode

The external reference mode uses external reference sources to recover the signal of interest inside the HF2 Instrument. In this mode, the internal frequency generators are not used to stimulate the DUT. As the signal reference is an arbitrary periodic signal, a certain amount of time is required for the HF2LI to lock on the reference and to be able to recover the signal of interest reliably. This lock time depends on several parameters, but most important on the level and phase noise of the reference.

The external reference mode is supported with single-channel and dual-channel operation. This is possible as the HF2 Instruments includes 2 independent measurement units that are working autonomously. In single-channel mode, the reference can be fed into the Input 2/Ref connector on the front panel. This alternative provides an unmatched capability to use references with small amplitudes as they can be amplified by the signal path of Input 2. In dual-channel operation, the external TTL references are fed into the HF2 by means of the DIO0 and DIO1 connectors on the back panel.

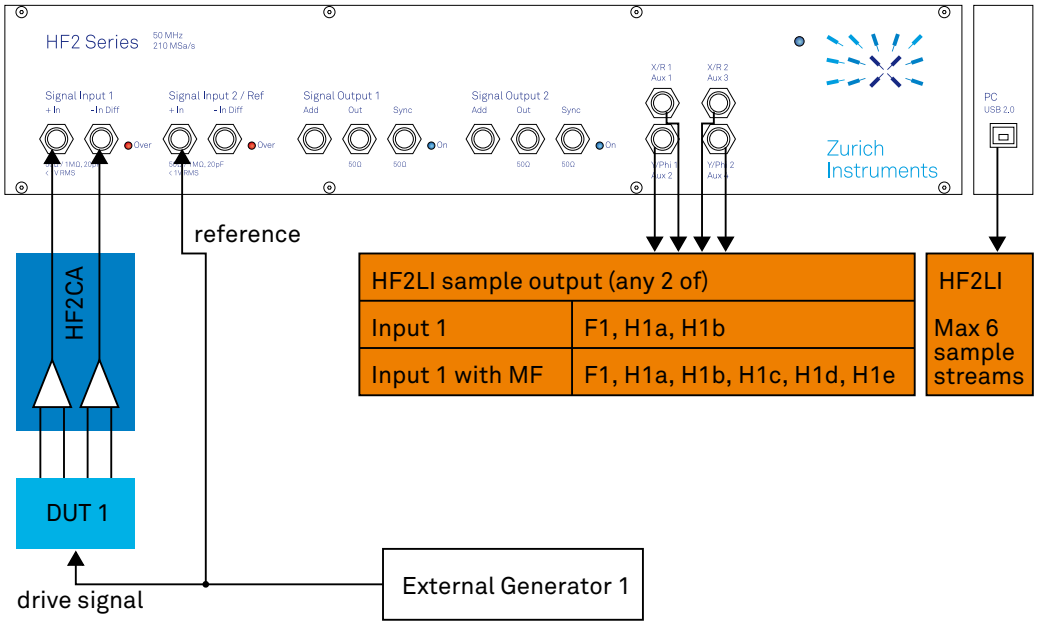


Figure 2.7. HF2 external reference mode / single-channel

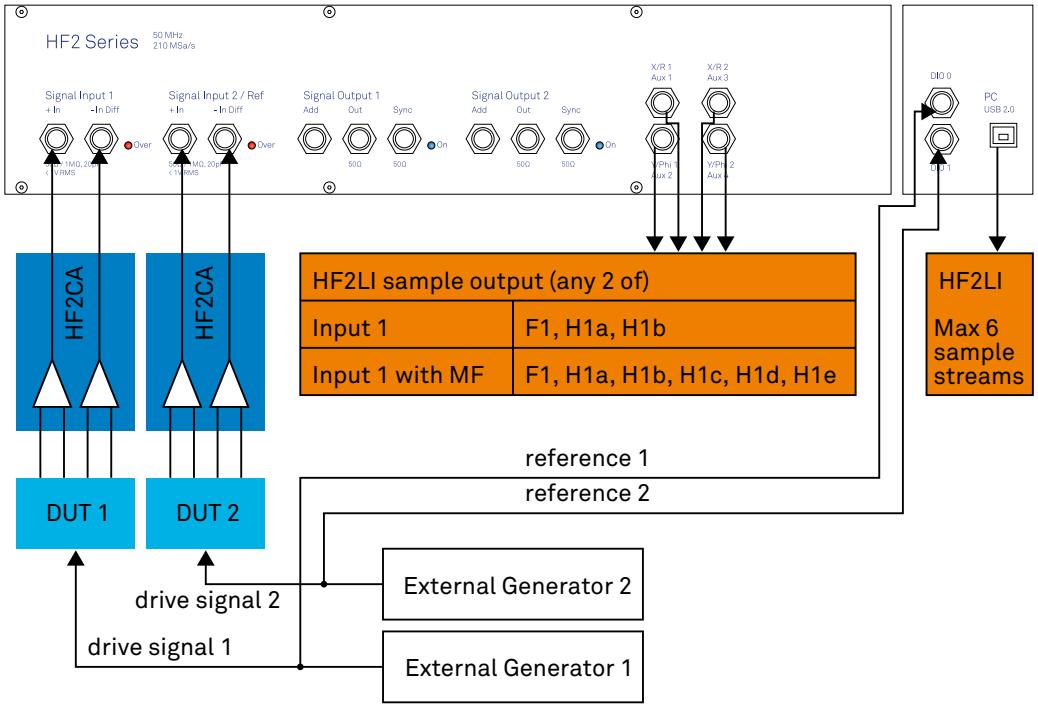


Figure 2.8. HF2 external reference mode / dual-channel

2.5.3. Auto Reference Mode

The auto reference mode makes use of the internal PLLs to recover the reference frequency directly from the signal coming from the DUT. In this mode, the internal frequency generators are not used to stimulate the DUT. As the reference is inherently contained in the sampled signal, a dedicated PLL is able to lock on the frequency and to recover the reference and the signal of interest. This process is suited for signals with enough amplitude and signal-to-noise ratio. Further the reference recovery requires a certain amount of time that depends on several parameters like the level and the phase noise of the measured signal.

The auto reference mode is supported with single-channel and dual-channel operation. This is possible as the HF2 Instrument includes 2 independent measurement units that are working autonomously. In dual-channel mode it is sufficient to connect the signals captures at the DUTs to the Input 1 and Input 2 connectors of the HF2 Instrument. The HF2 Series support both single-ended and differential input signals ideal for fixed and floating ground applications.

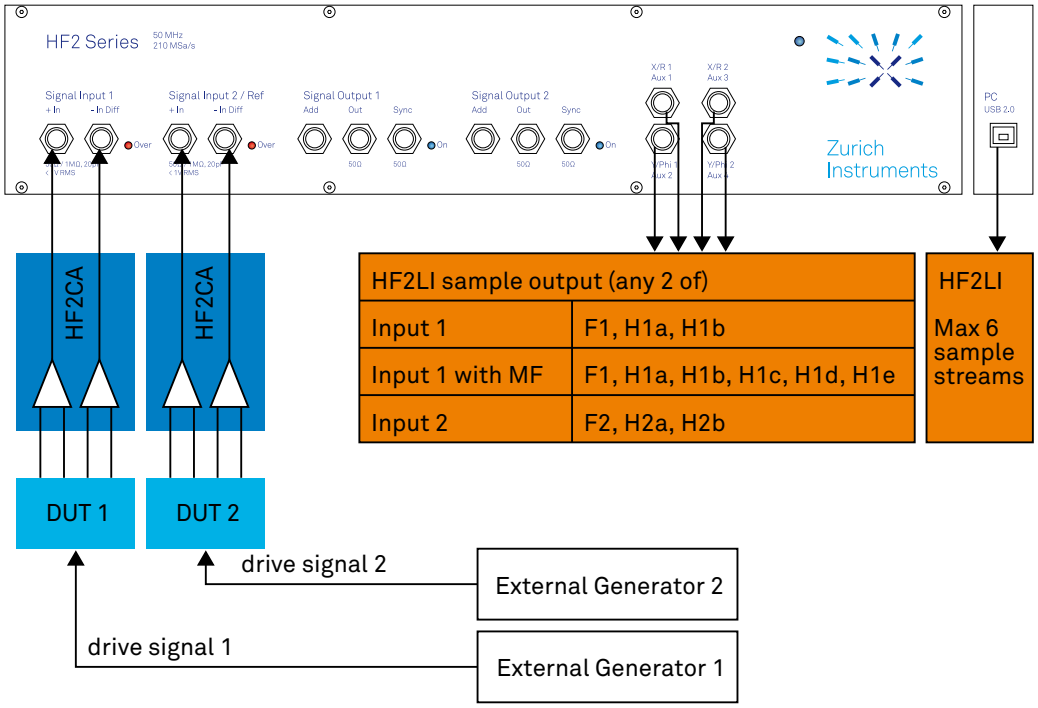


Figure 2.9. HF2 auto reference mode / dual-channel

2.5.4. Multi-frequency Operation

The multi-frequency operation is the powerful extension provided by the HF2 Series increasing the number of frequencies that can be analyzed in parallel. Moreover, the multi-frequency considerably expands the multiplexing options the user has with respect of input channels and demodulator clocks. Please note that the HF2IS-MF is different than the HF2LI-MF (see Table 2.4) as different features and different number of demodulators are activated.

For the HF2LI the multi-harmonic mode and the arbitrary frequency mode are distinguished. In multi-harmonic mode it is possible to analyze a signal at the fundamental frequency and at 5 harmonics at the same time, and the arbitrary frequency mode is the extension to analyze a signal of interest at 6 completely independent frequencies.

For the HF2IS only the arbitrary frequency mode is relevant.

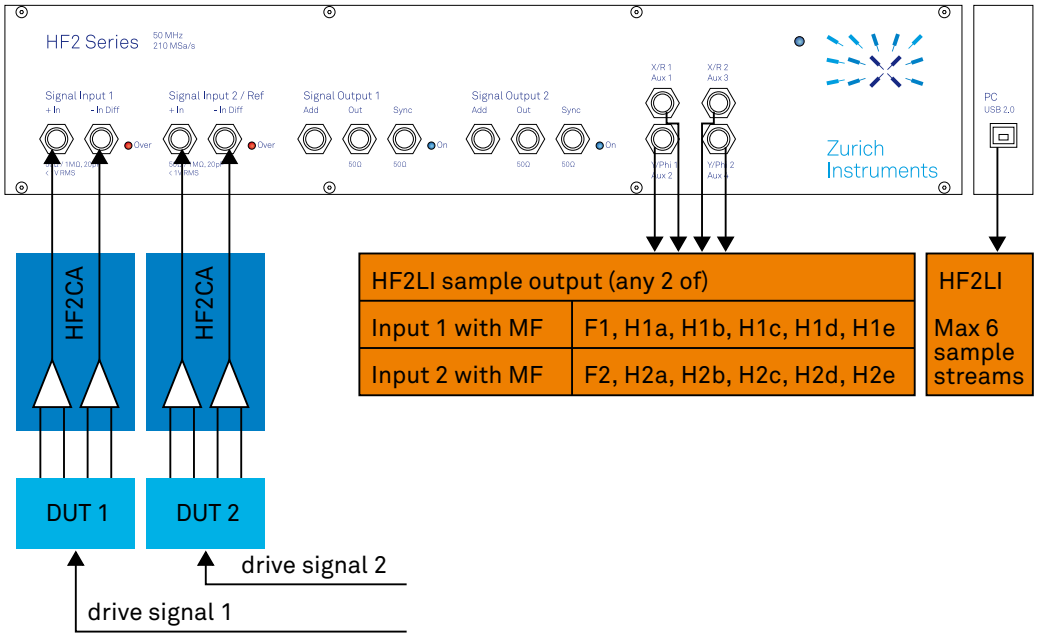


Figure 2.10. HF2 multi-harmonic mode

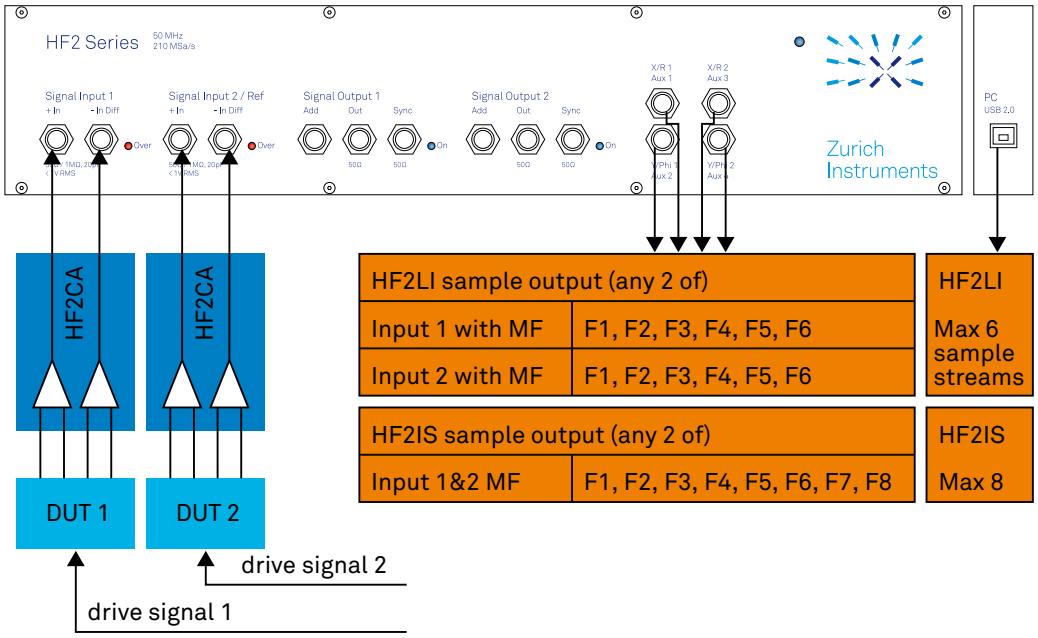


Figure 2.11. HF2 arbitrary frequency mode

Chapter 3. Tutorials

The tutorials in the chapter have been written to permit users becoming familiar with lock-in amplification, with the concept of host-based lock-in amplifiers, and the graphical user interface of the HF2 Instrument, ziControl. The tutorials require basic laboratory equipment and equipment handling knowledge. For the tutorials, you need the following material:

- 1 USB 2.0 cable (supplied with your HF2 Instrument)
- 3 BNC cables (2 optional)
- 1 male shorting cap (optional)
- 1 oscilloscope (optional)
- 1 T-piece (optional)

Note

For all tutorials, you must have the LabOne and ziControl packages installed as described in the Getting Started Chapter . Further the ziServer and ziService installed by the LabOne must be running on your computer. If you are unsure, check in the Windows Task Manager that the `ziServer.exe` and `ziService.exe` tasks are running (make sure you show processes for **all** users). If you are using Linux, use the command `ps -ef ¦ grep ziServer`. In case the Server is not running you find information on how to start it in the Software Installation Section . Finally, run the graphical user interface ziControl from the Windows Start Menu or using the Linux command `ziControl &`.

3.1. Tutorial HF2LI First Time User

This tutorial covers the basic operations for the HF2LI lock-in amplifier and for ziControl.

The ziControl graphical user interface is provided as the primary interface to the HF2LI but it is not the only program that can run the instrument. Typically, the user will use ziControl to set up the instrument and then either use ziControl to take the measurements or run (possibly concurrently) some custom programs. The ziControl window is divided in two sections, the Settings section on the top and the Tools section on the bottom (see also the Graphical User Interface Overview Section). The Tools section is common for all users whereas the Settings section differs according to the number of installed options.

3.1.1. The Lock-in Tab

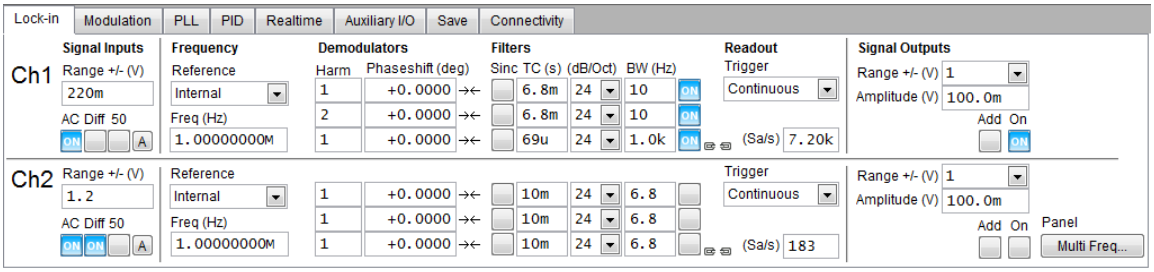


Figure 3.1. The Lock-in tab

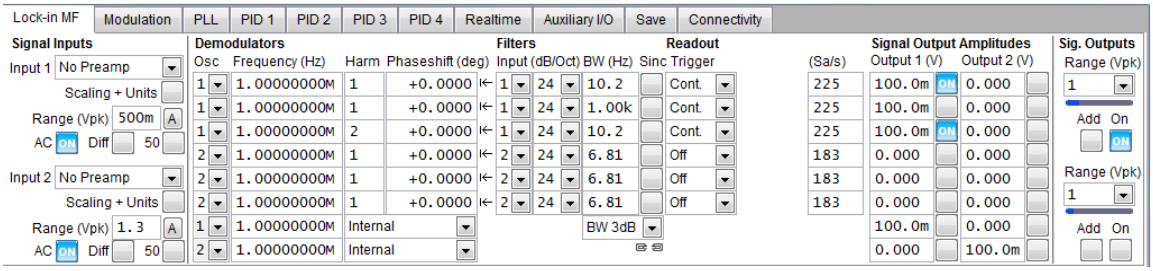


Figure 3.2. The Lock-in MF tab

Click the Lock-in tab in the settings section of the user interface. If you find the Lock-in MF tab instead, it means that the instrument has the HF2LI-MF option installed. In this case the Lock-in tab is not accessible. It is not possible to switch between the two tabs because the settings influence each other. Section 4.2.1 provides the full documentation of the Lock-in tab while Section 4.2.2 describes is detail the Lock-in MF tab.

The Lock-in section is split into Ch1 to control Signal Input 1/Signal Output 1 and Ch2 to control Signal Input 2/Signal Output 2. The two channels are in all aspects equivalent: in this tutorial we will consider Ch1. The Signal Input section contains a Range that can be set to a value between 1 mV and 1.6 V, the largest amplification of the input signal is achieved for 1 mV. The input has protection diodes that clip signals with amplitude above 5 V.

Important

Please respect the compliance to the maximum ratings Table 9.2 to prevent damage to the instrument.

The AC/DC button sets the coupling type: AC coupling has a cutoff frequency of 1 kHz. The AC coupling consists of a blocking capacitor between two input amplifier stages: this means that a DC signal larger than 5 V will saturate the front amplifier even if AC coupling is enabled. The Diff button sets single ended/differential measurements mode: in the differential mode, the voltage difference between the +In and -In is amplified. The 50 Ω button toggles the input impedance between low (50 Ω) and high (approx 1 MΩ) input impedance. 50 Ω input impedance should be selected for signal frequencies above 10 MHz to avoid artifacts generated by multiple signal reflections within the cable. With 50 Ω input impedance, one will expect a reduction of a factor of 2 in the measured signal if the signal source also has an output impedance of 50 Ω. The A (Auto range) button automatically sets the input range to be twice the input signal.

Next, one finds the reference signal/demodulation section. For the demodulation, the lock-in also needs a reference signal. The reference source can be

- internal: the lock-in internal signal generator will produce the reference/excitation signal, or
- external: the reference is generated externally and supplied to the lock-in. Possible choices include the Signal Input 1 (auto reference mode), in which case the phase information is of course zero, Signal Input 2, in which case the phase information is retained; in these two cases, the bandwidth for the reference signal is the full bandwidth, 50 MHz. Other choices are the two Analog Inputs and two Digital Input/Output (DIO) located on the rear side of the HF2. The bandwidth for the Analog inputs is 20 kHz, the bandwidth for the DIO is 2 MHz (the DIO accepts a TTL signal with lows/highs of <0.8 V/>3.3 V).

For the purpose of this tutorial, set the reference to Internal and the frequency to 1 MHz.

Under the section Demodulators the user can select which harmonics and filter bandwidths to use for demodulation. It is not uncommon to need to measure different harmonics (integer multiples of the fundamental frequency, in this case 1 MHz). Select the first harmonic to 1 for the first demodulator (the first line), set the filter order to 24 dB/oct (this is 80 dB/dec, an attenuation of 10^4 for a tenfold frequency increase) and type 10 Hz into the BW control (the digital filters of the HF2 are described in Section 10.3). Users are sometimes interested in the second harmonic that may be generated by non-linear processes in their device under test: select harmonic 2 for the second demodulator and type the same values for the filter order and BW as in the previous case. You can also measure the same fundamental harmonic with a larger bandwidth: first make sure that the chain symbol is unlinked by clicking on it if necessary (otherwise the newly typed settings will be copied to all demodulators), then set harmonic to 1, order to 24 dB/oct and BW to 1 kHz. Measuring with different bandwidths can provide the signal average and transient values. Click on the enable button next to the filters to read out the values from the 3 demodulators.

Next, set the Trigger to Continuous and the Rate to 7.20 kSample/s (rate settings can only be sub-multiples of 460 kSample/s, the maximum readout rate for one demodulator): in this case, the HF2LI will send the demodulated signal sampled at this rate through the USB. Due to the finite bandwidth of the USB connection the maximum cumulative demodulator sample rate is 700 kSample/s, which can be split over the active demodulators, see Table 9.4. In this example we're using 3 active demodulators, therefore, since the sample rates are sub-multiples of 460 kSamples/s the maximum possible readout rate for each demodulator is 230 kSample/s. Note that, according to the Nyquist sampling theorem , the sampling rate should be at least twice as fast as the maximum frequency present in the signal, if order to reconstruct the demodulated signal (this is not important if you only need one data point or the standard deviation of the demodulated signal). Since the low pass filters do not have an infinite roll-off (the attenuation is not infinite past the filters 3 dB point), it is common to set the sampling rate to about 8 times higher than the filter bandwidth.

Next, in the output section, select the excitation amplitude to 100 mV and the output range to be the smallest possible but at least twice as large as than the amplitude for minimum harmonic distortion. Connect Signal Output 1 to Signal Input 1 +In with a BNC cable and click on the On button in the ziControl Signal Outputs section.

3.1.2. The Numerical Tool

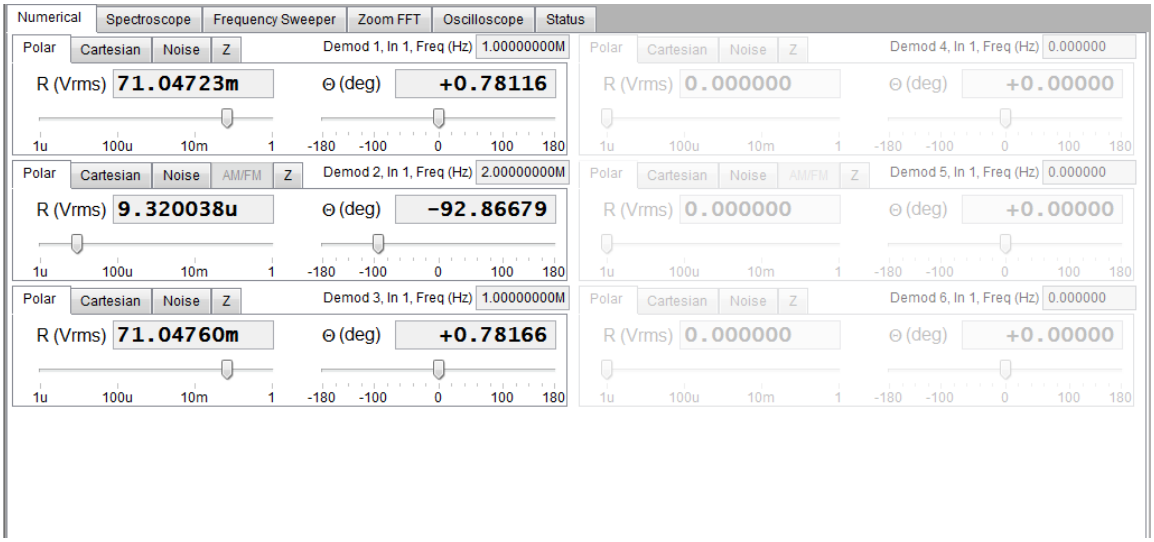


Figure 3.3. The Numerical tab

In the Numerical tab located in the Tools section, you should read 71 mV RMS for the R component of demodulator 1, (demodulating at 1 MHz). The RMS corresponds to the 100 mV divided by √2. The phase value will depend on the BNC cable length (for lengths shorter than one meter, the phase is approximately a few degrees). Demodulator 3 (also at 1 MHz) will show the same amplitude, but the digits fluctuate more, since the measurement bandwidth and therefore the noise, is larger. Demodulator 2 reads only a few µV because at 2 MHz (the second harmonic) there is only a little component of the signal, coming from the harmonic distortion of the HF2LI output and input stages.

3.1.3. The Spectroscope Tool

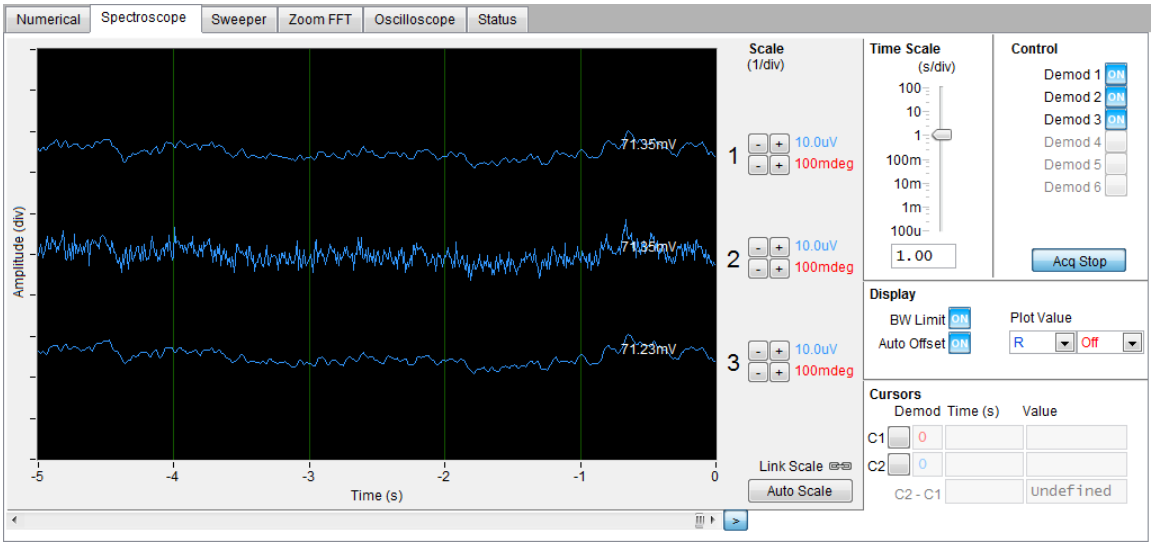


Figure 3.4. The Spectroscope

Now click the Spectroscope tab in the tools section. Here one can display the demodulated values over time. Select demodulators 1, 2 and 3, make sure that Acq Stop is not pressed, and then press the Auto Scale button. The demodulated traces for these three demodulators are displayed, offset to each other: as before, demodulators 1 and 3 have the same average value, but a larger noise amplitude is clearly visible in the third trace. If the Auto Offset button is enabled, an offset is continuously subtracted from the traces, so the traces are displayed in the middle of the panel. The vertical axis V/Div and deg/Div can be set individually by the +/- buttons for each demodulator. The vertical scale adjustment can also be linked together through Link Scale button. The horizontal axis scale s/DIV can be set through the Time Scale slider. One can inspect the older data by dragging the time slide below the horizontal axis: the amount of stored data depends on the computer memory (the higher the sample readout rate, the shorter will be the displayed trace). To display the most recent data, click the greater than symbol next to the slide. Please note that the displayed plot suffers from aliasing if the demodulator operates with wide filters, the sample readout rate is large, and the trace is zoomed out to span a long temporal interval: this effect is due to the limited resolution of the computer screen versus the effectively available samples - enabling the BW Limit partially solves this problem by adding another filter on the displayed samples.

3.1.4. The Oscilloscope Tool

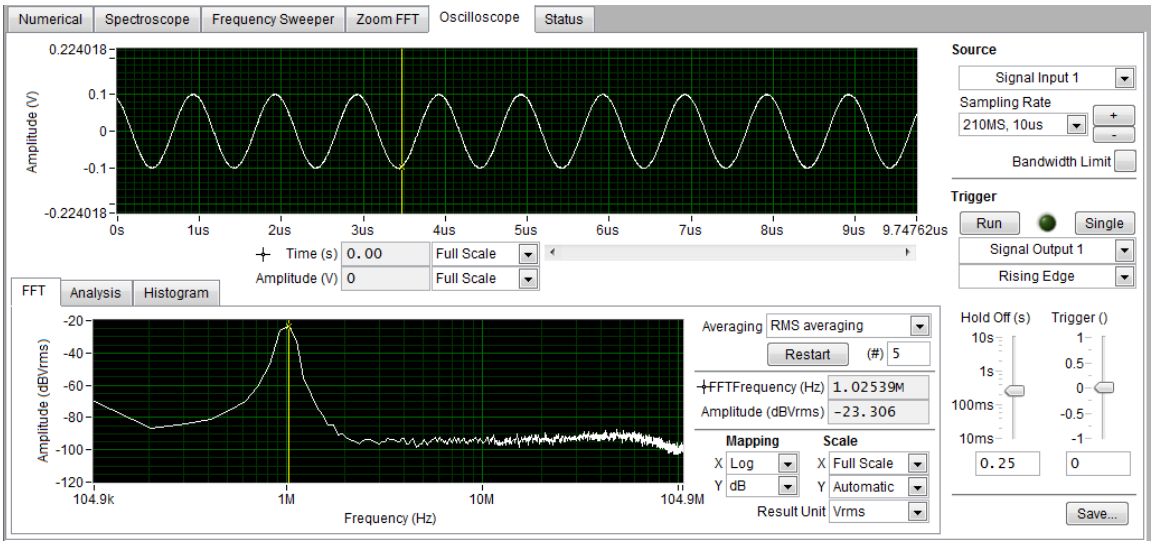


Figure 3.5. The Oscilloscope

Let us proceed to the Oscilloscope tab. This displays the digitized signal at the signal inputs and outputs of the HF2LI. It is a 2048 point wave trace that is useful for visualizing the signal to demodulate; it also replaces the need for an external oscilloscope. Select Signal Input 1 from Source and Signal Output 1 from Trigger and press the Run button. The 1 MHz input signal is visible as 10 full cycles if the Sampling Rate is set to 210 MS, 10 µs. Decreasing the sampling rate to display a longer time interval should be done carefully because it may lead to undersampling (i.e. aliasing due to violation of the Nyquist criteria): for instance setting the sampling rate to 26 kSample/s, 80 ms, will produce a correct looking sinusoidal, but at the wrong frequency. The Bandwidth Limit button may reduce aliasing effects without removing them completely.

The update rate of the oscilloscope frames is controlled by the Hold Off slider: the minimum interval between two traces is 10 ms. This is a low value which increases the load of the USB

bandwidth and may lead to USB sample loss - therefore avoid using small hold off values if not needed.

A Fourier transform of the trace is displayed in the graph below: the frequency resolution is coarse because the time trace contains 2048 points. Averaging of the Fourier power spectra can be enabled to increase the SNR ratio. Moreover, the FFT result is available as spectral density (power per unit bandwidth) by selecting the Result Units $V_{RMS}/\sqrt{Hz}$ or V_{RMS}^2/Hz. The oscilloscope trace can be saved in ASCII format with the Save button.

3.1.5. The Sweeper Tool

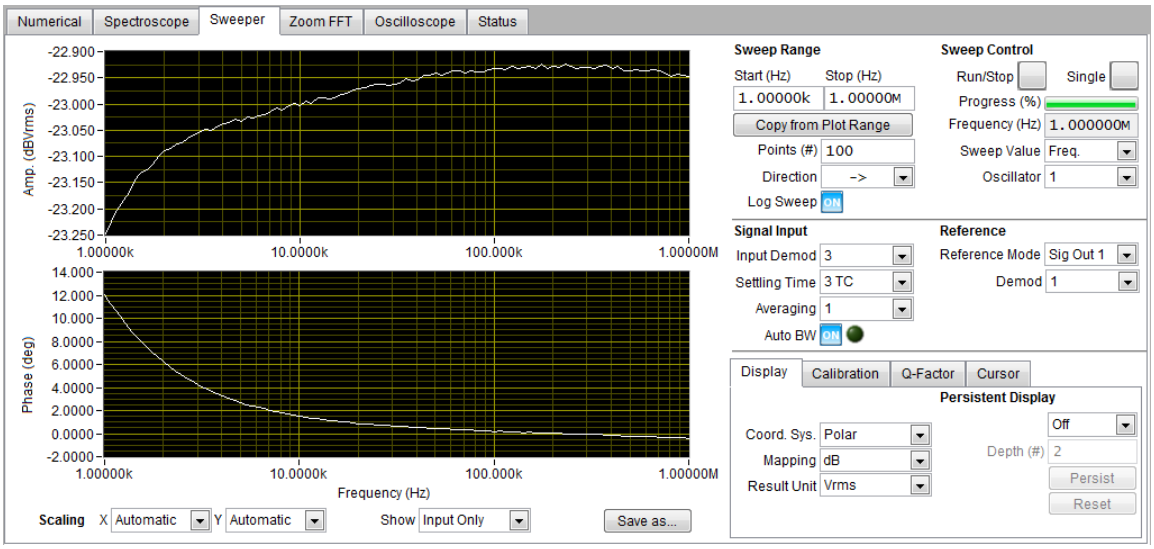


Figure 3.6. The Sweeper

Next is the Sweeper tool: it turns the HF2LI into a network analyzer, giving the transfer function of a device under test in the form of a Bode plot. AFM and resonator users find this useful to easily identify the resonance frequency of their cantilevers as well as the phase delay. The sweeper tool can also be used to sweep parameters other than frequency : phase, time constant, amplitude and auxiliary output.

As a frequency sweeper example, we will test a logarithmic sweep of 100 points between 1 kHz and 1 MHz. Set the Sweep Range Start to 1 kHz and Stop to 1 MHz , 100 points and enable the Log Sweep. Select the reference to be output 1, output 2 or one of the 6 demodulators. This will allow the user to observe the ratio between the measured value and any of the selected reference signal. (Note that one is now able to get a ratio between two different demodulators.) Click on Run/Stop for continuous sweeping or on Single for a single sweep. Toggle the AC input coupling in the Lock-in settings, and observe the attenuation in the response around 1 kHz in AC coupling, since the AC coupling has a cutoff frequency of approximately 1 kHz. You will notice that despite the excitation signal is 100 mV, the Amplitude (upper graph) shows 1 (or 0 dB) when Input/Ref is selected in the Show pull-down list: the sweeper displays the transfer function of the device under test by dividing the measured signal by the user-defined reference signal. One can also show the absolute measured input value only by selecting Input Only or the reference value by selecting Ref Only. The Bode plot that is displayed can be saved pressing the Save button - the data file in ASCII format contains 8 columns. The first 6 columns are F (Hz), R (V_{RMS}), Θ (deg), NEPBW (Hz), R_ref(V) and Θ_ref (deg). The remaining two columns are the display variables and units that are chosen under the Display tab. The Settling Time pull-down allows the user to control the amount

of time between setting at a new frequency and taking a reading: during this time, the filter will settle towards its final value. 1 TC means a minimum (guaranteed) settling to 90%, 3 TC to 95% and 10 TC to 99% of the total signal change. Minimum guaranteed settling is specified because the USB latency will add to the wait time.

During the logarithmic sweep the NEPBW (noise equivalent power bandwidth) is adjusted for each frequency point and displayed under the Filters BW field under the Lock-in tab. The adjustment is due to the fact that the sweep is logarithmic and the sweep frequency steps are not equally spaced. In order to account for all signal power (and power densities), the measurement bandwidth must be changed accordingly. This can be done automatically through the Auto BW button. For an explanation of the NEPBW, see Signal Processing chapter. Note that in this configuration, if the signal (at the modulation frequency) to background noise ratio is large, there will not be any effect when disabling Auto BW, since the noise signal is negligible when measured with (almost) any NEP bandwidth. Averaging can also help to improve the signal-to-noise ratio during the sweep.

As an example of noise measurement, disconnect the BNC cable from Signal Output 1 and connect it to Signal Output 2. In the Lock-in tab, turn off the Signal Output 1, and generate a 100 kHz / 100 mV excitation Signal Output 2 (remember to turn on the output in the Signal Outputs section). In the frequency sweeper perform a single sweep with Auto BW enabled. A relatively wide peak will appear at 100 kHz, as the measurement was performed with wide NEPBW. Switch the X scaling to Manual and select the zoom tool next to the graph and zoom in the region around 100 kHz; click the Copy from Plot Range button to use the new boundaries for the sweep as selected in the graph and again perform a single sweep. The peak at 100 kHz will appear narrow, reflecting the change in the measurement bandwidth. Note that the preferred way of measuring noise is to use the zoomFFT tool described in Section 3.1.6 and Section 10.6).

3.1.6. The zoomFFT Tool

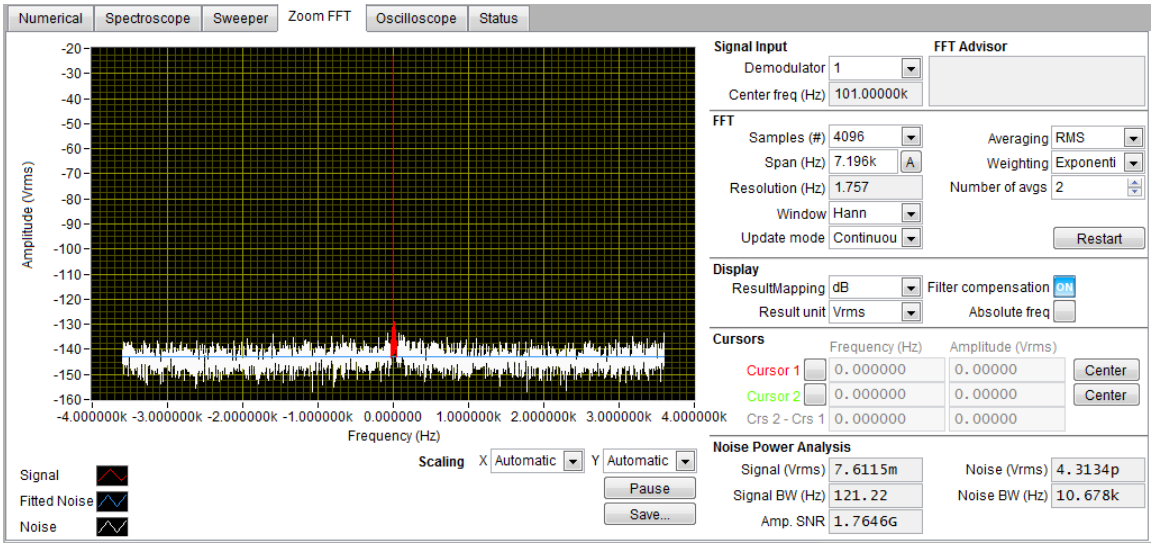


Figure 3.7. The zoomFFT tab

The zoomFFT tool (more information (see Section 10.6) allows the user to measure the frequency spectrum around a specific frequency: this is done by performing the Fourier transform of the demodulated X and Y (or in-phase and quadrature) components of the signal (more precisely of the quantity X+jY, where j is the imaginary unit). The reason is that the demodulation process

shifts the spectrum of the input signal by the demodulation frequency and the Fourier transform of the demodulated X+jY corresponds to the frequency spectrum of the input signal around the demodulation frequency. zoomFFT and FFT coincide when the demodulation frequency is zero. The frequency resolution that can be achieved in this way is given by the sampling rate divided by the number of recorded samples, and is therefore much higher than the frequency resolution obtained in the Oscilloscope tab. The zoomFFT approach is more efficient than the FFT on raw samples in which one digitizes a long time trace, performs the Fourier transform and retains only the portion of the frequency spectrum of interest while discarding the rest.

We continue from the previous section with the BNC connecting Signal Output 2 to Signal Input 1, and 100 mV, 100 kHz sine wave. In the Lock-in tab, type 101 kHz in the reference frequency for Ch 1, 48 dB/oct, 500 Hz bandwidth, a Readout sampling rate of 7.2 kSample/s and Filter compensation enabled. The high order filters and sampling rate compared to the bandwidth are so chosen to avoid aliasing. In the zoomFFT tab, select Demodulator 1 for Signal Input. A peak appears at 1 kHz to the left of the center frequency. Increasing the number of lines in FFT will result in a finer frequency resolution. The Filter compensation button compensates for the demodulator filter, by dividing the FFT spectrum by the demodulator filter transfer function: that is why the input signal does not appear attenuated despite being outside the filter bandwidth (1 kHz and 500 Hz respectively). Filter compensating facilitates the job of researchers who want to find the noise background level.

3.1.7. The Status Tool

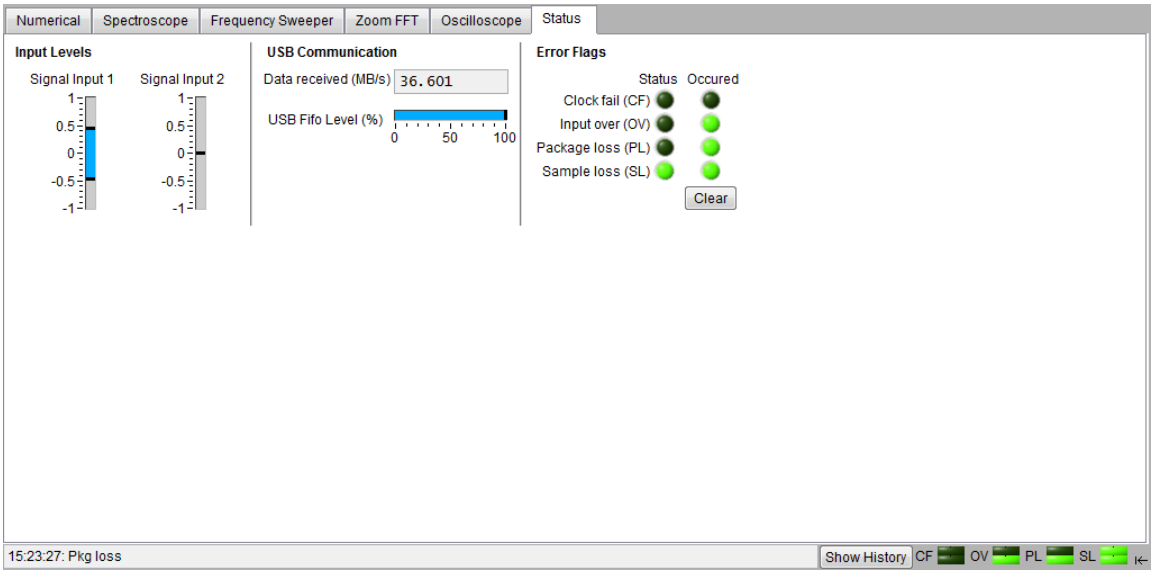


Figure 3.8. The Status tab

In the Status tab, several general status settings are displayed, such as the input signal displayed as percentage of the full input range, the USB FIFO level and input overloads. For instance, setting the Readout rate to a value of 230 kSample/s and turning on 3 or more demodulators will saturate the USB bandwidth and result in the Sample loss (SL) warning LED going on; data points are dropped by the USB subsystem and lost. To prevent data loss, a smaller readout rate or fewer demodulators should be used simultaneously. Other error flags are also displayed here, such as Input overload (OV, at the same time a red LED will turn on the front panel of the HF2LI) and Clock Fail (CF) if an externally supplied 10 MHz reference clock (not the external reference frequency for demodulation) does not meet the requirements in terms of amplitude or frequency. Once

the cause for the warnings has been identified and the solved, the Status LED goes off but the Occurred LED still stay on to remind the user that an abnormal situation occurred in the past: all status flags can cleared by using either the Clear button or the arrow at the bottom right corner of ziControl.

3.1.8. The Auxiliary I/O Tab

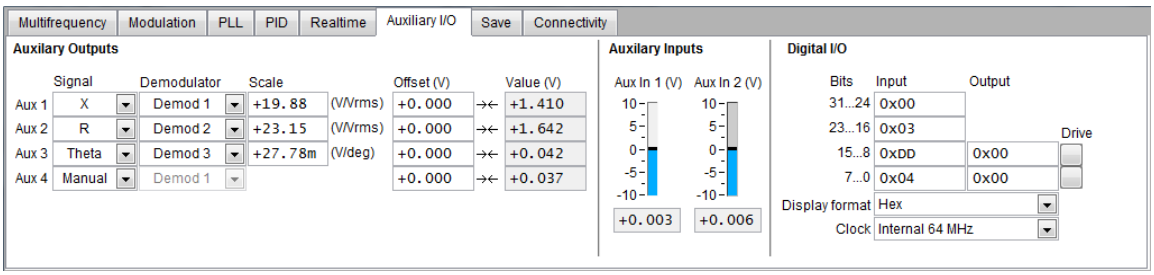


Figure 3.9. The Auxiliary I/O tab

The Auxiliary I/O tab in the settings section controls the 4 Auxiliary Outputs on the right side of the HF2LI front panel, as well as the 2 Auxiliary Inputs and the DIO connector on the rear side. In the Lock-in tab, set the Reference to Input 1 (Auto): the HF2LI finds the correct input signal frequency at 100 kHz (this illustrates the use of the external reference). On the Auxiliary I/O, output the signal amplitude by selecting R from the Signal pull-down menu from Aux 1 and Demod 1 from Demodulator. Set the Scale factor to 10 V/V_{RMS}: you should read 0.712 V in the output Value (V) tab, which corresponds to the amplitude of the signal as you can read it in the Numerical tab, multiplied by the scale factor. If one is interested in small variations of the signal amplitude, an offset can be applied to the output: type -0.712 in Offset (V) (or alternatively click the counter facing arrows next to Offset (V)): Value (V) should now read 0.

3.1.9. The Save Tab

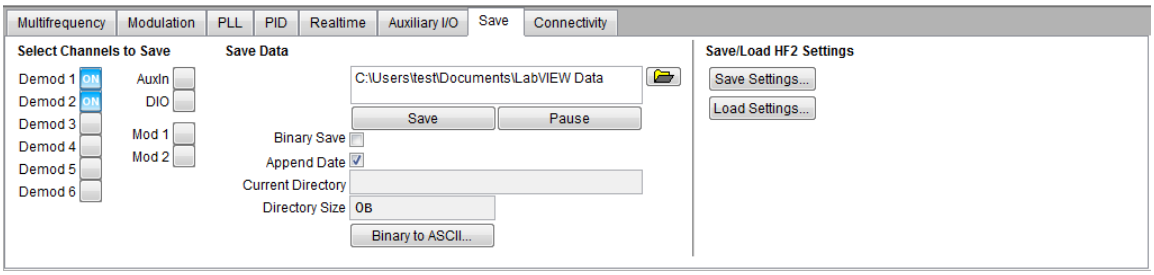


Figure 3.10. The Save tab

In the Save tab, the user can save data from multiple demodulators into a directory whose path can be specified by clicking on the folder icon. Select the two demodulators you want to save, Demod 1 and Demod 2 from Select Channels to Save (make sure that demodulators 1 and 2 are active in the Lock-in tab), select Append Date (this will create a directory with the timestamp

appended to the save directory base name to avoid overwriting a pre-existing save directory) and click on the Save button. Data from demodulators 1 and 2 will be saved at the readout rate programmed in the Lock-in tab. To stop saving data, click again on the Save button.

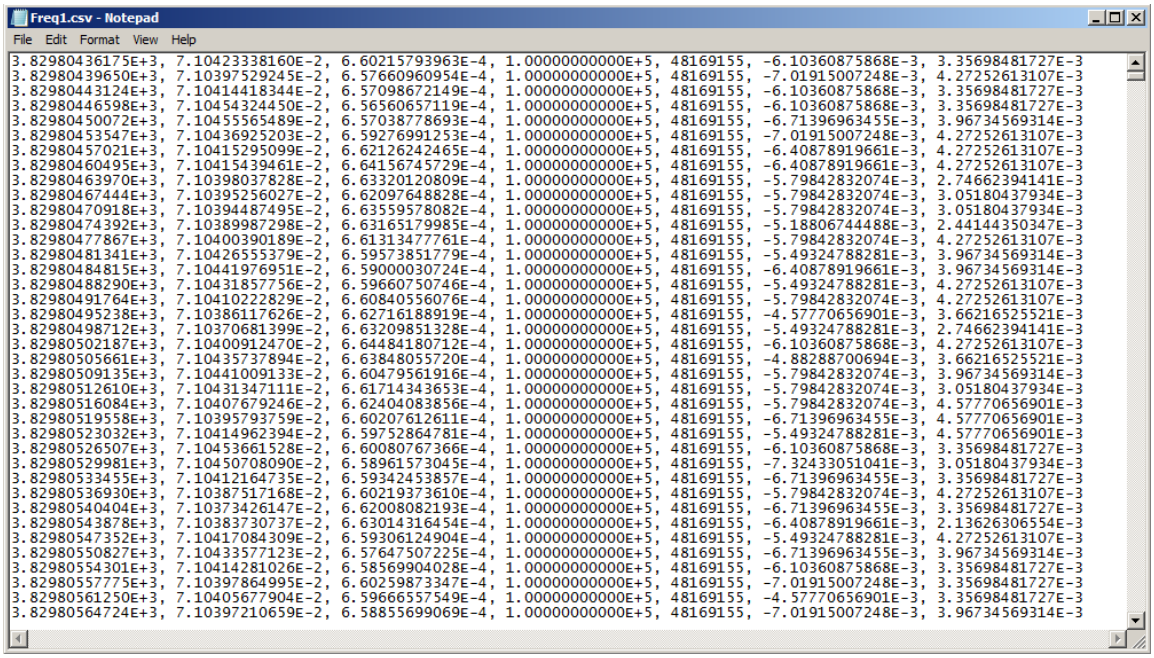


Figure 3.11. The file containing the saved demodulated samples in ASCII format: the 7 columns relate to Timestamp [s], X [V_{RMS}], Y [V_{RMS}], Frequency [Hz], DIO value [decimal], AUX In 1 [V], AUX In 2 [V]

The newly created directory contains a CSV file for each saved demodulator, which you can read with any text editor if the file was saved in ASCII format (Binary Save not selected). If fast readouts are needed, binary save is an option for high performance save without the overhead for the conversion to ASCII. The data can be converted to ASCII at the end of the save with the Binary to ASCII button.

The HF2LI settings can be saved to file and reloaded (for instance after power-up of the HF2LI). Since settings are saved in ASCII format as node-value pairs, the settings can be inspected and modified using a normal text editor. For more information of how instrument settings are organised in the node-tree, see the Section 6.1.3 .

3.1.10. The Connectivity Tab

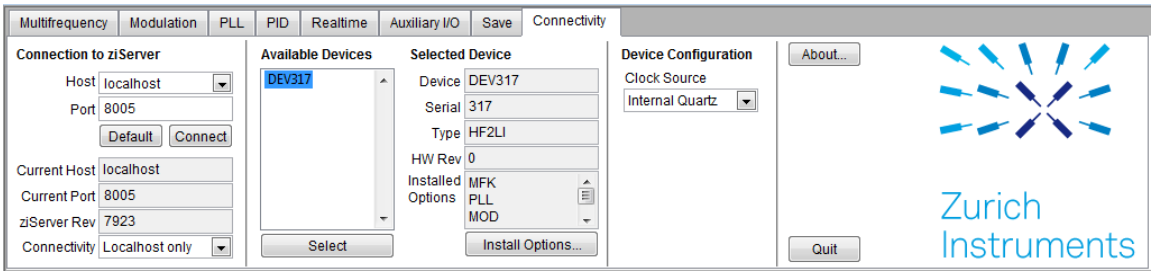


Figure 3.12. The Connectivity tab

The Connectivity tab show all the devices connected to the computer and additional information on the HF2LI that is selected, such as the options installed. When multiple devices are available, the user may select the device to be controlled with the current instance of ziControl. The connectivity tab also shows that ziControl connects to ziServer, via TCP/IP by making a loop-back connection to the local computer at port 8005. ziServer manages all communications between the HF2LI and all programs that want to access the HF2LI, such as ziControl and any user program, and it therefore must be always running.

Figure 3.13. The Command Line display

3.2. Tutorial Simple Loop

3.2.1. Preparation

In this tutorial you generate a signal with the HF2 Instrument and measure that generated signal with the same instrument. This is done by connecting Signal Output 1/Out with Signal Input 1/+In with a BNC cable. This tutorial shows a single-ended operation, meaning that there is no signal going into the Signal Input 1- connector. For proper operation, the Channel 1 must be set to single-ended operation, or alternatively the Input 1 - connector must be shorted to ground using a male shorting cap. Optionally it is possible to connect the generated signal at Output 1 to an oscilloscope by using a T-piece and an additional BNC cable.

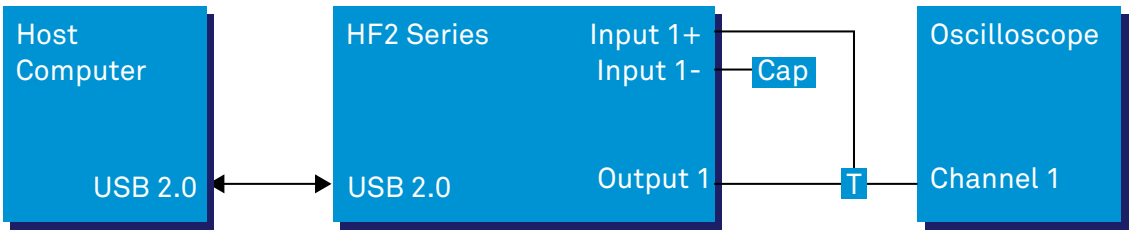


Figure 3.14. Tutorial simple loop setup

Note

This tutorial is both for HF2LI lock-in amplifier and HF2IS impedance spectroscope users.

Connect the cables as described above. Make sure the HF2 unit is powered on, and then connect the HF2 to your computer with a USB 2.0 cable. Finally launch the ziControl application (Start Menu/Programs/Zurich Instruments/ziControl).

3.2.2. Generate the Test Signal

Perform a number of settings in order to generate a 2.5 MHz signal of 0.5 V amplitude on the Output 1 connector.

- Go to the Lock-in tab and set frequency of Channel 1 to 2.5 MHz: click on the field, enter 2.5M or 2.5E6 and press <TAB> on your keyboard to confirm the data
- In the Output 1 section, set the Range pull-down of 1 V
- In the Output 1 section, set the amplitude to 0.5 V by entering 0.5 followed by a <TAB>
- By default all physical outputs of the HF2 are inactive to prevent damage to connected circuits. Now it is time to turn on the main output switch by clicking on the button labeled "On". The switch turns to blue indicates now "ON"
- If you have an oscilloscope connected to the setup, you are able to see your generated signal

Table 3.1. Settings: generate the test signal

Output range Ch1	1 V

Oscillator frequency Ch1	2.5 MHz
Oscillator amplitude Ch1	0.5 V
Output Ch1	ON

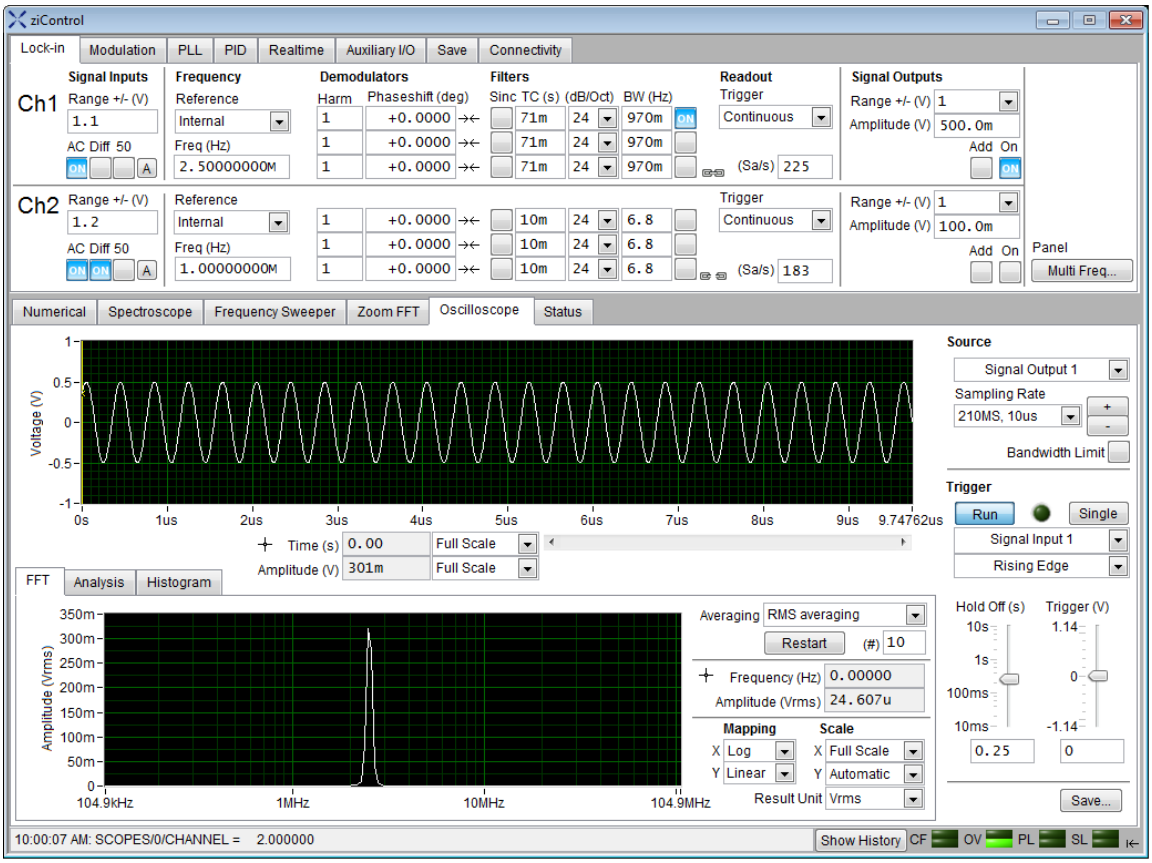


Figure 3.15. ziControl displaying the generated signal

3.2.3. Acquire the Test Signal

Next, you adjust the input parameters in order to detect signals with the right input range. To do this, you switch the signal source and the trigger of the oscilloscope to Input 1. Then you make sure that the input signal does not exceed the set range accordingly.

Table 3.2. Settings: acquire the test signal

Oscilloscope source	Input 1
Oscilloscope trigger select	Signal Input 1
Oscilloscope time scale	210MS, 10us
Oscilloscope trigger	RUN
Signal Input Ch1 range	1 V (approximated to 0.98 V)
Signal Input Ch1 AC/Diff/50	ON/OFF/OFF

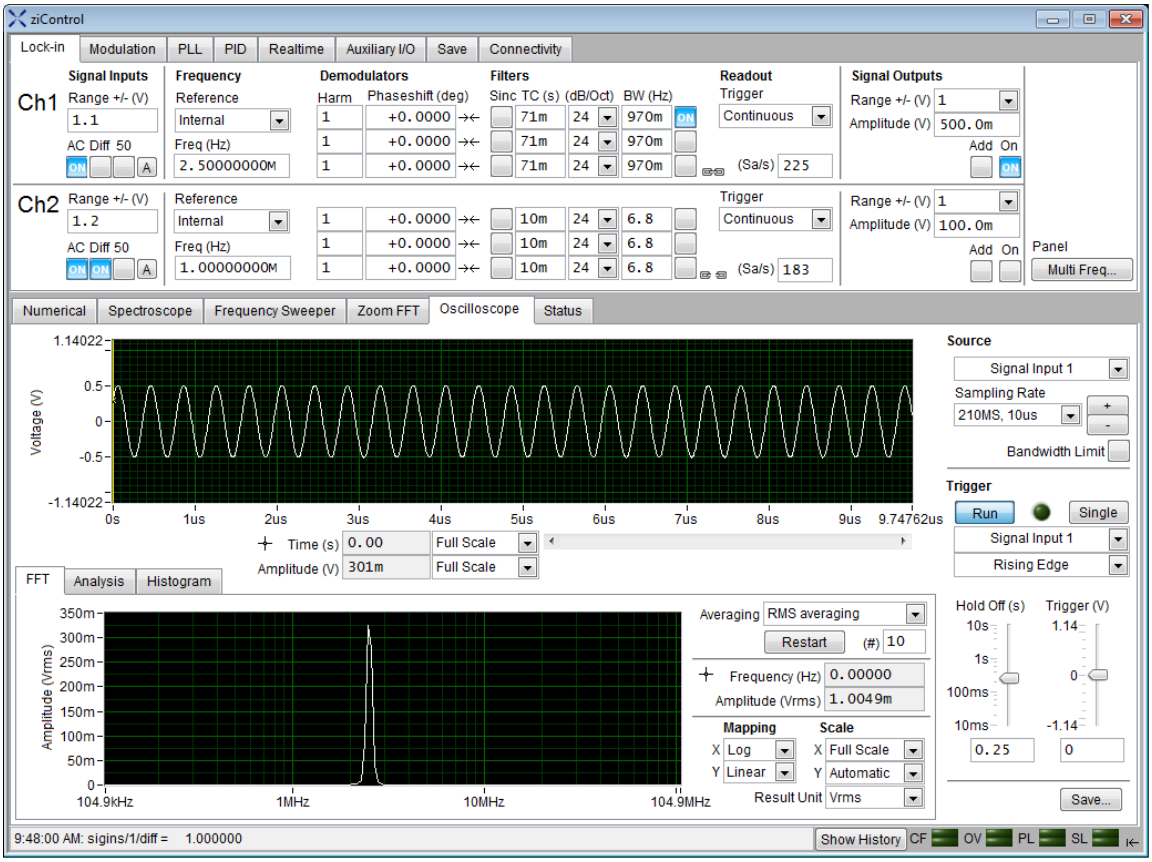


Figure 3.16. ziControl displaying the acquired signal

The oscilloscope tool displays the measured signal at Input 1. Having set the input range to 1 V ensures that no signal clipping occurs. If you set the input range to 0.3 V you see the effect on the scope window. Note how the red "Over" LEDs on the front panel of the HF2 indicates the error condition and the set OV status flag on the right-bottom corner of the window. Clear the flag by pressing |←, and set back the input range to 1 V.

The oscilloscope is a very handy tool to quickly check the settings before proceeding to more advanced measurements. For the full description of the oscilloscope tool please refer to oscilloscope description .

3.2.4. Measure the Test Signal

Next you use the demodulators of the HF2 to measure acquired test signal. You will use 2 tabs from the tools section of the graphical user interface: the Numerical and the Spectroscope tools.

First set the following settings (choose any of the 6 available demodulators).

Table 3.3. Settings: measure the test signal

Time constant (TC)	10 ms (approximated to 10.2 ms)
Filter slope	12 dB/Oct
Resulting measurement bandwidth (BW)	~10 Hz

Demodulator readout rate	100 Hz (approximated to 112 Hz)
Demodulator enable	ON

These settings set the demodulation filter to second-order low-pass operation with a 10 ms time constant. The corresponding bandwidth is calculated automatically according to the formula explained in Chapter Functional Description . The output of the demodulator filter is read out with 100 Hz, implying that 100 data samples are sent to the host PC per second. These samples are viewed in the numerical and spectroscope tool that we examine next.

The numerical tool provides the space for 6 measurement panels corresponding to the 6 demodulators. Each of the panels has the option to display the samples in Cartesian (X,Y) or polar format (R,Θ) together with the set frequency. The unit of the (X,Y,R) values are V_{RMS}. The numerical panels are a little psychedelic as many digits are toggling at the same time. This could be annoying at first, but consider that you are measuring thermal noise and that the units are small (µV range, or depending on the filter settings nV). This provides a first glimpse of the accuracy of the measurements performed by your HF2 Instrument.

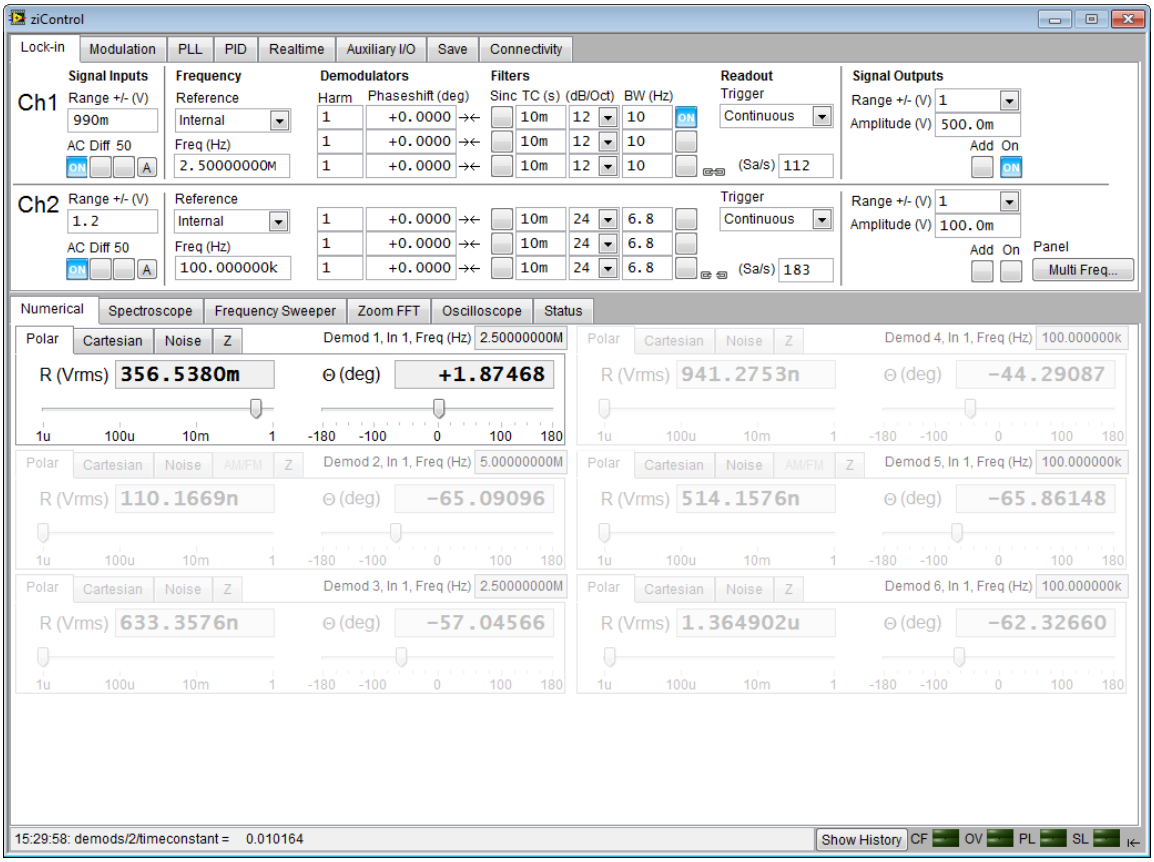


Figure 3.17. ziControl numerical view with 1 active demodulator

If you wish to play around with the settings, you could now change the amplitude of the generated signal, and observe the effect on the demodulator output.

Next you have a look at the Spectroscope tool. This tab provides a time plot of the demodulator outputs. It is possible to plot up to 6 signals continuously as (X,Y) or (R,Θ) pairs, set different scales, or make detailed measurements with 2 cursors. For detailed explanation of the spectroscope spectroscope description .

First operate the following settings, and later you may use different settings of the sliders to obtain a different view. In order to set the scales, press the related "+" and "-" buttons.

Table 3.4. Settings: plot the demodulated test signal

Time scale slider	10.4 s/DIV
Signal scale	20 µV/DIV
Phase scale	100 m°/DIV
Display settings	R and Theta
BW Limit / Auto Offset switches	ON/ON
Enable demodulator switch	ON

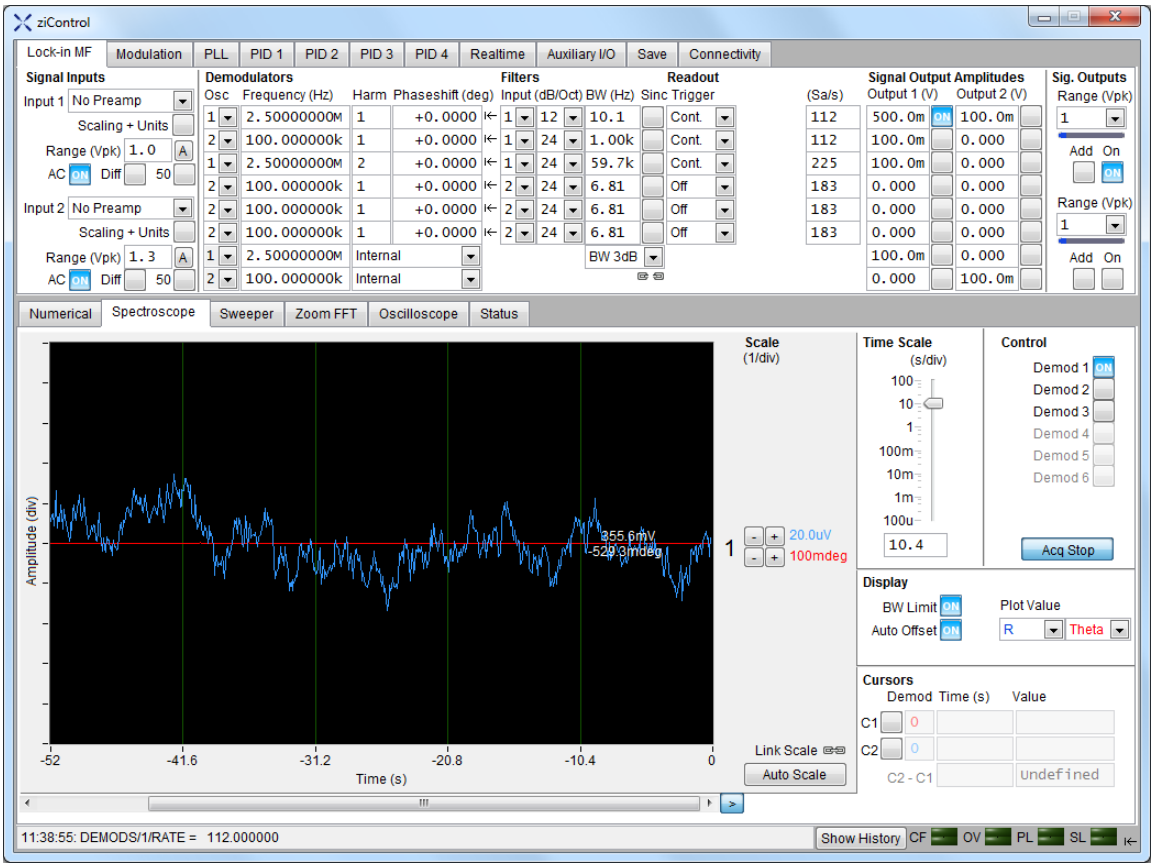


Figure 3.18. ziControl spectroscope view with plot of demodulator 1 output

You may try different scale settings. The "Auto Scale" button might also help to find suited scale settings for (X,Y,R,Θ).

3.2.5. Different Filter Settings

As last step of this tutorial you change the filter settings and see the effect on the measurement results. For instance you change the time constant of the integration to 2 seconds.

Table 3.5. Settings: changing the filter settings

Time constant (TC) per order	2 s
Filter slope	12 dB/Oct

Resulting measurement bandwidth (BW)	~51 mHz

Increasing the time constant, increase the integration time of the demodulators smoothing out the demodulator outputs. This averages the noise over time and the output of the filters toggles much less.

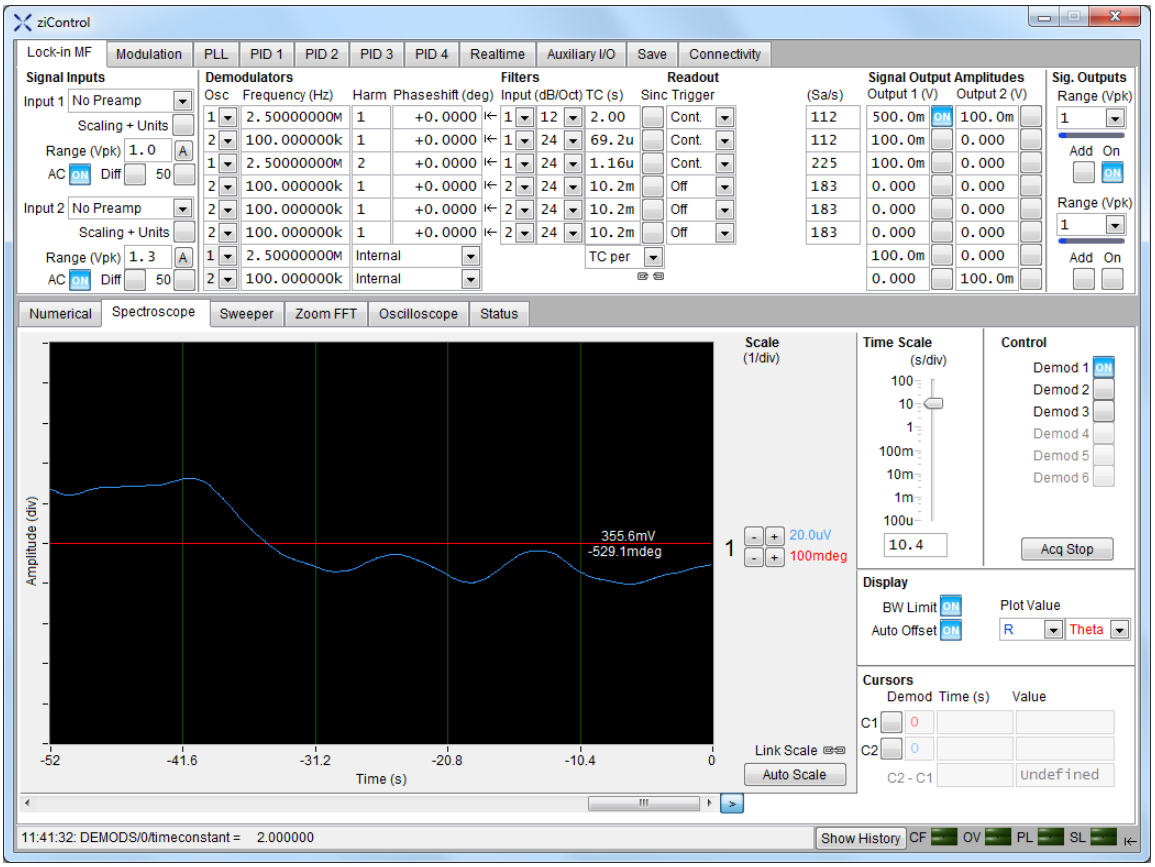


Figure 3.19. ziControl spectroscope view with plot of demodulator 1 output (TC = 2 s)

Further you may for instance "disturb" the demodulator with a change of test signal amplitude, from 0.5 V to 0.7 V and vice-versa. The blue plot seems to skyrocket, which you can control by pressing the "Auto Scale" button. With high time constant, the demodulators change much slower to changing input signals compared to low time constants. The number of stable digits in the numerical tool is also higher with high time constant.

3.3. Tutorial Dynamic Signals

3.3.1. Preparation

In this tutorial you generate a test signal of 2.5 MHz with an amplitude oscillating at 1 Hz. Then you measure the test signal using 2 different filter settings.

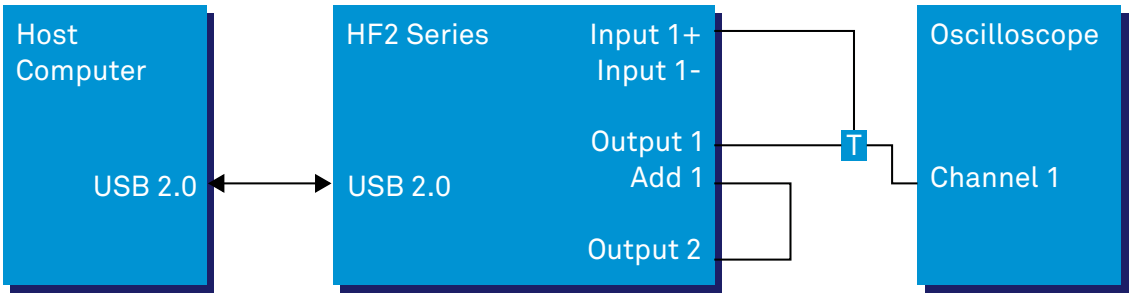


Figure 3.20. Tutorial dynamic signals setup

Note

You must have installed the LabOne and ziControl packages as described in the Getting Started Chapter , and the ziServer must be running on your computer. If you are not sure about that, check the in the Windows Task Manager whether there are the **ziServer.exe** and the **ziService.exe** tasks running. Alternatively check the ziServer as described in the Programming Chapter .

Note

This tutorial can be performed both on the HF2LI lock-in amplifier and on the HF2IS impedance spectroscope and will use the Input connector. The generation of multi-frequency signals is simple on the HF2LI with the HF2LI-MF option or on the HF2IS, where there is no need to make use of the ADD connector.

Connect the cables as depicted above. Make sure the HF2 unit is powered on, and then connect the HF2 to your computer with a USB 2.0 cable. Finally launch the ziControl application (Start Menu/Programs/Zurich Instruments/ziControl).

3.3.2. Generate the Test Signal

In this section you generate a 2.5 MHz sinusoidal signal whose amplitude oscillates at 1 Hz. This is also called the beat signal. In order to obtain this test signal you add (or superimpose) two sinusoids of the same amplitude but with a 1 Hz difference in the frequency.

Table 3.6. Settings: generate the test signal

Output range Ch1	1 V

Oscillator frequency Ch1	2'500'000 Hz
Oscillator amplitude Ch1	0.3 V / ON
Oscillator frequency Ch2	2'500'001 Hz
Oscillator amplitude Ch2	0.3 V / ON
Output ADD Ch1	ON

When connecting an oscilloscope to the Output 1 connector, you should be able to observe the superposition of the 2 sinusoids. To see the acquired signal inside the graphical user interface ziControl switch to the oscilloscope tool and The ziControl oscilloscope view looks like this with the following settings.

Table 3.7. Settings: acquire the test signal

Oscilloscope source	Signal Input 1
Oscilloscope trigger select	Signal Input 1
Oscilloscope sampling rate	6.4 kS, 320 ms
Oscilloscope trigger	RUN
Signal Input Ch1 range	1 V (approximated to 0.98 V)
Signal Input Ch1 AC/Diff/50	ON/OFF/OFF

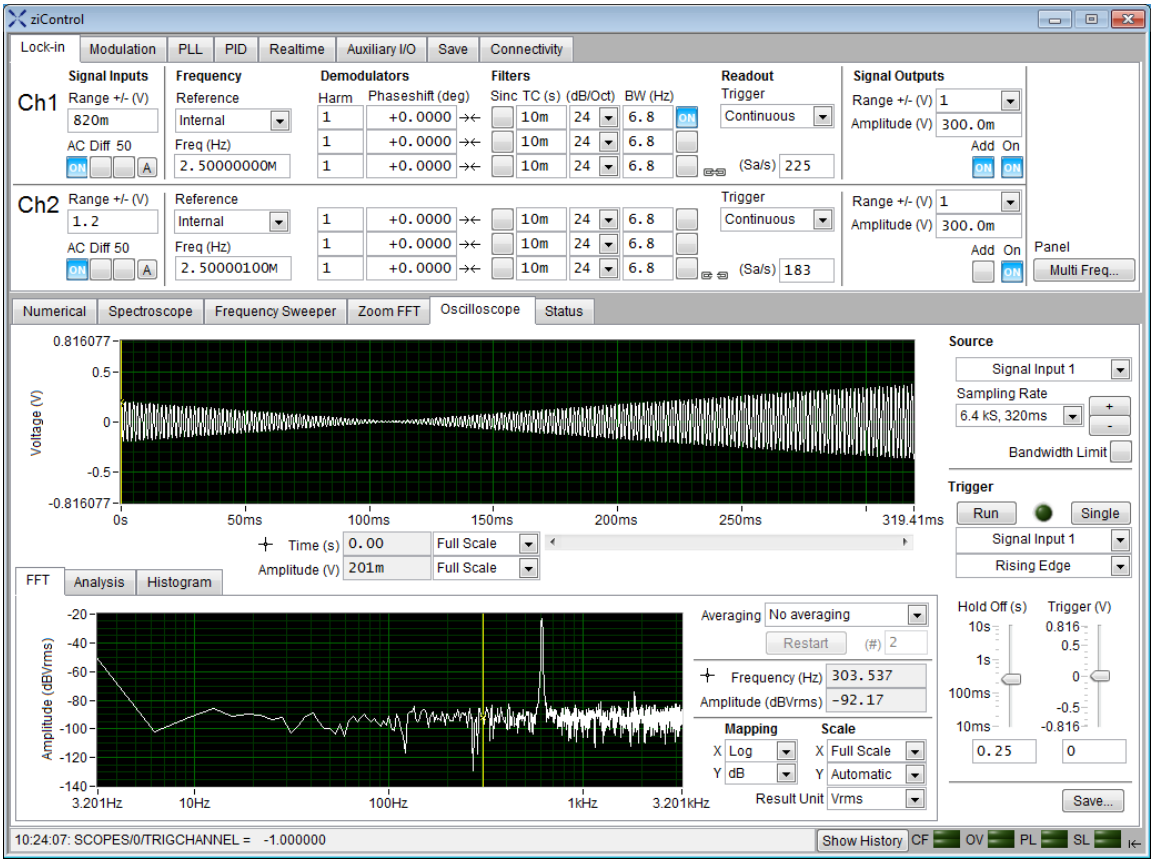


Figure 3.21. ziControl displaying the acquired signal

The beat signal has a maximum amplitude of 0.6 V, thus within the set range of 1 V. The range setting will prevent any higher voltage than what is set - even if 2 sinusoidals of 0.7 V amplitude each would be added like done in this section, the output would be clipping at 1 V which is the

set range. Try to change the output range to 0.1 V, and see how the output voltage is changed to prevent inconsistent settings.

3.3.3. Measure the Test Signal

First you change to the spectroscope tool, set the scale in order to view an interesting set of data, and set the demodulator filters to a low time constant to measure the amplitude of the 2.5 MHz signal (Hull curve).

Table 3.8. Settings: filter with a low time constant

Time constant (TC)	10 ms (approximated to 10.2 ms)
Filter slope	12 dB/Oct
Resulting measurement bandwidth (BW)	~10 Hz
Demodulator readout rate	100 Hz (approximated to 112 Hz)
Time scale slider	1.58 s/DIV
Signal scale	100 mV/DIV
Phase scale	20 °/DIV

These settings set the demodulation low pass filter to a 10 ms time constant (the corresponding bandwidth is around 10 Hz) and the filter slope to second order. The output of the filter read out with 100 Hz, samples which are sent to the host computer.

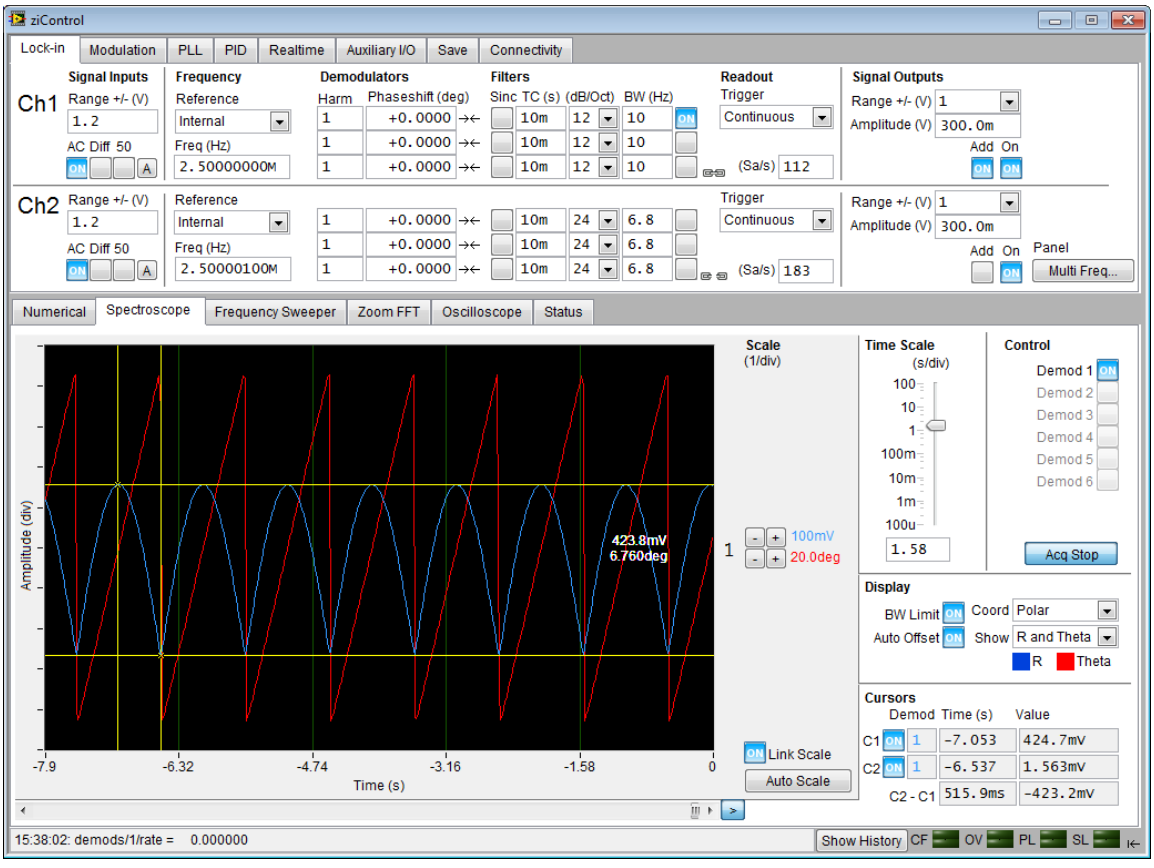


Figure 3.22. ziControl displaying the measured signal

If you stop the acquisition by pressing the button "Acq Stop" you can conveniently measure the amplitude of the 1 Hz signal by using the 2 cursors C1 and C2: 394.4 mV_{RMS}, half period 498.1 ms. You can achieve higher measurement precision by using a even lower time constant (e.g. 1 ms), increasing the readout rate (1.8 kHz), and zooming into the spectroscope view using a different time scale.

Next you use a high time constant to separate the 2 original sinusoids even though they are superposed in one signal. Keep the spectroscope tool, and in the lock-in tab apply the following settings.

Table 3.9. Settings: filter with a high time constant

Time constant (TC)	2 s
Filter slope	24 dB/Oct
Resulting measurement bandwidth (BW)	~35 mHz
Demodulator readout rate	100 Hz (approximated to 112 Hz)

These settings set the demodulation low-pass filter to a time constant of 2 s, with a resulting measurement bandwidth of 35 mHz. With these settings the HF2 is able to distinguish between the signal at 2'500'000 Hz and the signal at 2'500'001 Hz as the measurement bandwidth is considerably less than the spacing of the 2 test signals. The output of the demodulator is stable after a setting time.

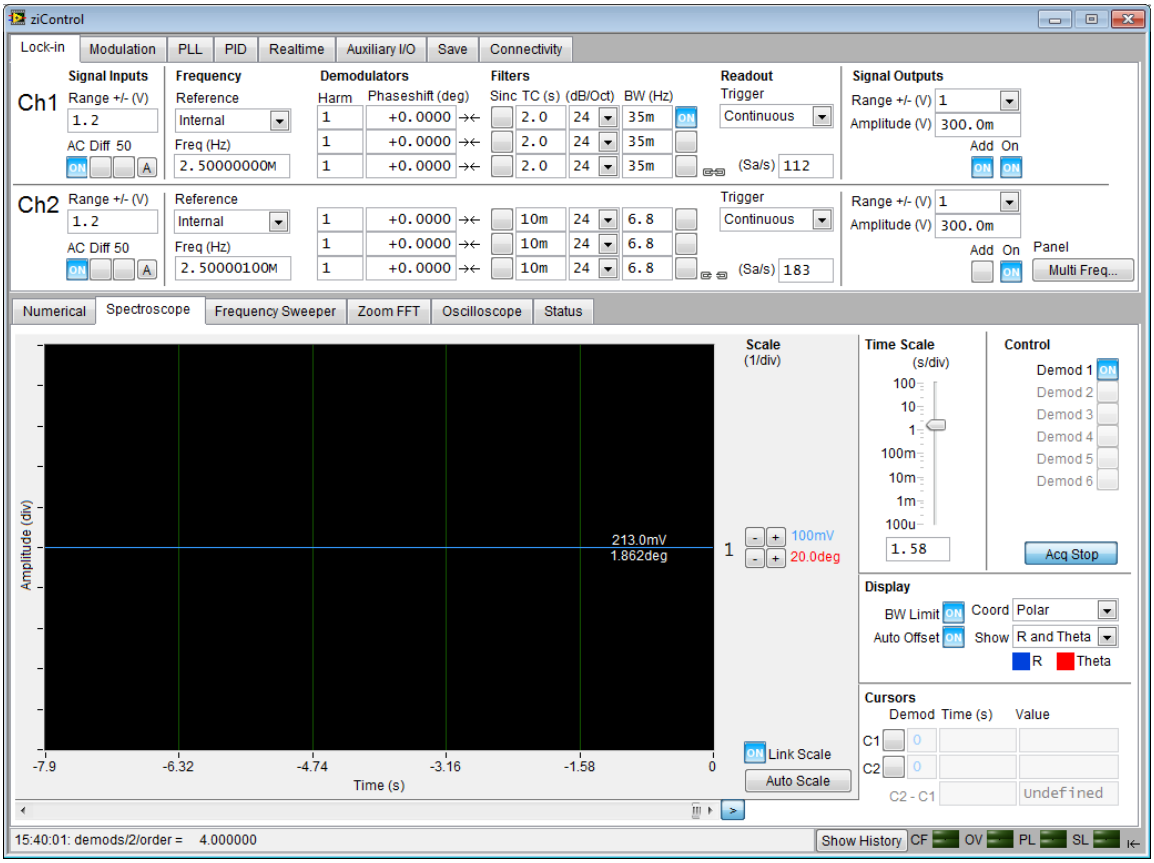


Figure 3.23. ziControl displaying the measured signal

The output of the demodulator does not show any oscillations like before: the numerical value is 212.8 mV_{RMS} with an angle of 53.57°. If you switch to the oscilloscope view, you see that the signal

at Input 1 is still beating as before, while the demodulator filter is set such to ignore the interferer at 2'500'001 Hz. Try to switch-off the interferer.

Table 3.10. Settings: remove the interferer

Signal output Ch2 switch	OFF

The numerical value changes to 212.8 mV_{RMS} with an angle of 53.6°. The time it takes to settle depends on various parameters like filter setting and switch-off timing. The difference in amplitude of the measurement at 2'500'000 Hz with or without interferer is in the range of 50 µV. With different filter settings it is possible to do better than that.

Consider this: you have 2 signals with relevant amplitude (0.3 V) interfering with each other as their frequency is very close (1 Hz at 2.5 MHz). The power of lock-in amplification consists of extracting the relevant signal energy at exactly one frequency. The "immunity" from nearby interferer is the capability to ignore them. This a simple definition of the dynamic reserve.

3.3.4. Filter Setting Discussion

This section aims to summarize a few basic concepts of filtering in connection with lock-in amplification. In this tutorial, you have used different filter settings to measure different signal properties.

Table 3.11. Settings: filter with a high time constant

Time constant	Measurement bandwidth	Noise measurement	Changes upon steady state change	Example
Low setting	High, capability to detect fast events	More noise in measurement result	Fast adaptation of result	BW = 10 kHz, capability to detect events at 2-5 kHz, prone to pick-up noise
High setting	Low, capability to determine the steady state	Less noise in measurement result	Slow adaptation of result	BW = 50 mHz, exact determination of steady state - events more frequent than 0.1 Hz are filtered

Filtering constitutes a trade-off between signal & noise pick-up and accuracy. In order to measure faster events, it is required to open up the filters allowing also more noise to influence the measurement result. The opposite is to measure with narrow filters increasing a lot the signal to noise ratio, but limiting the capability to detect the changes in the signal of interest. This trade-off is in common with any lock-in amplifier. The power of the HF2 is that it allows to do both at the same time thanks to the multiple demodulators per input channel.

3.4. Tutorial External Reference

3.4.1. Preparation

In this tutorial you generate a test signal of 1.0 MHz and use it as an external reference signal for demodulation.

This is done by connecting the Output 2 connector with the Input 2+ connector with a BNC cable. This tutorial shows a single-ended operation, meaning that there is no signal going into the Input 2- connector. Optionally, it is possible to connect the generated signal from Output 2 to an oscilloscope by using a T-piece and an additional BNC cable. The Output 1 connector is to be connected to the Input 1+ connector. This allows to check the regenerated reference signal. The measurement setup is shown in the following figure.

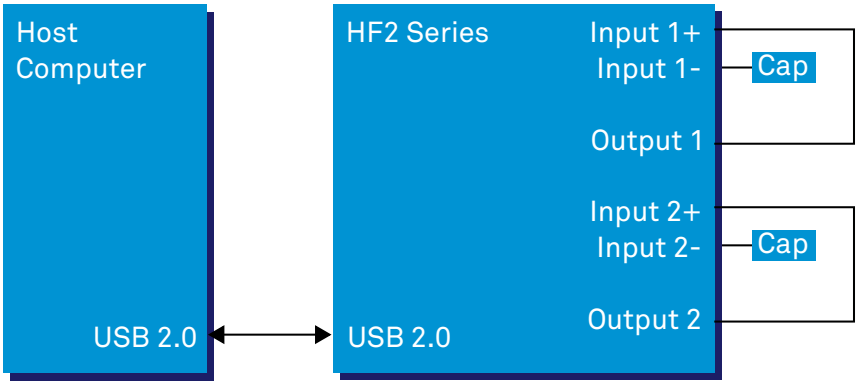


Figure 3.24. Tutorial external reference setup

You must have installed the LabOne and ziControl packages, and the ziServer must be running on your computer. If you are not sure about that, use the Windows Task Manager to check whether the **ziServer.exe** and the **ziService.exe** are running. Make sure the HF2 unit is powered on, and then connect the HF2 to your computer with a USB 2.0 cable. Finally, launch the ziControl application.

3.4.2. Generate the Test Signal

In this section you generate a 1.0 MHz signal with a 1 V amplitude on Output 2 for use as the external reference. The settings for generating the test signal are shown in the following table.

Table 3.12. Settings: generate the reference signal

Output range Ch2	1 V
Output amplitude Ch2	1.0 V / ON
Frequency Ch2	1 MHz
First demodulator Ch2	ON
Signal input range / AC / Diff / 50 Ch2	1.2 V / ON / OFF / OFF

When connecting an oscilloscope to the Output 2 connector, you should be able to observe the sinusoid. The ziControl oscilloscope view is shown in the following figure using the following settings.

Table 3.13. Settings: acquire the reference signal

Oscilloscope source	Signal Input 2
Oscilloscope trigger	Signal Input 2
Oscilloscope time scale	3

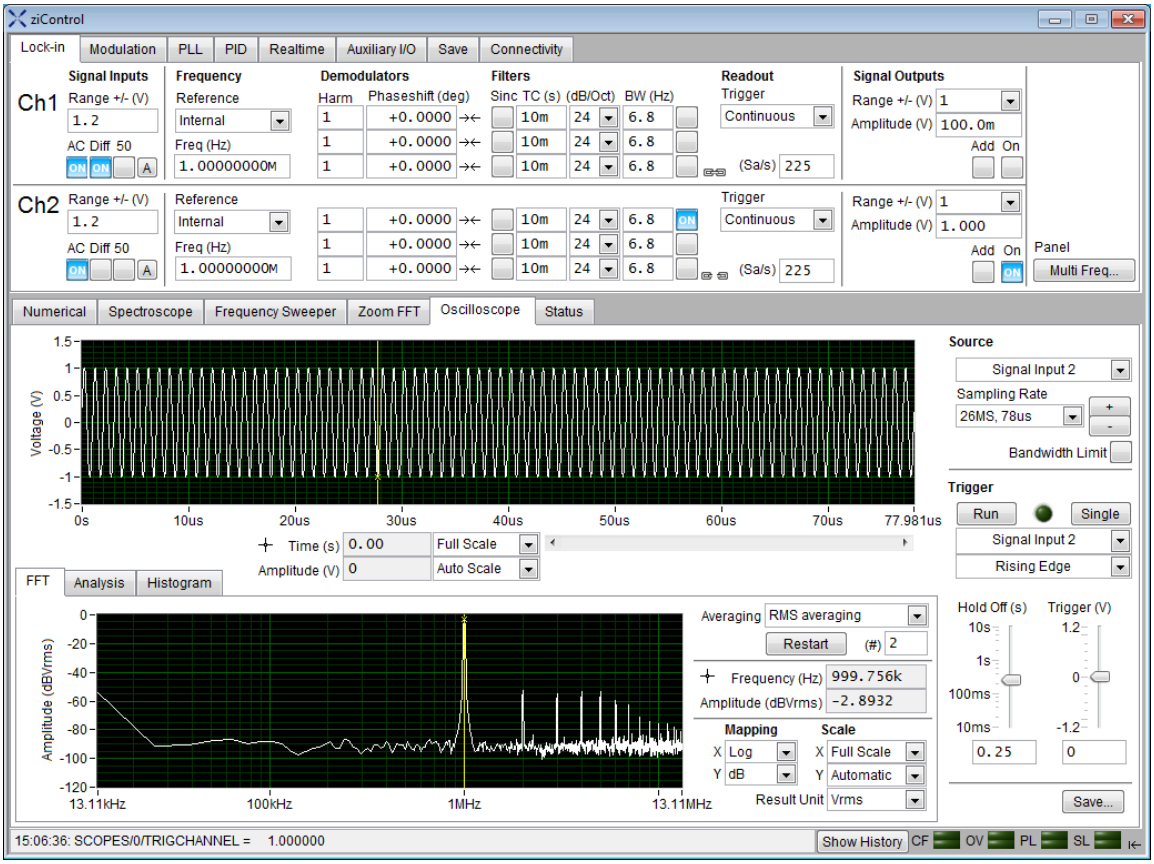


Figure 3.25. ziControl displaying the reference signal

Note

The FFT display can be used to check the frequency contents of the signal. Conveniently use the view settings automatic for the X axis and for the Y axis with logarithmic scale set (dB). The RMS averaging can be enabled to reduce the noise floor in the display.

3.4.3. Activate the External Reference Mode

In this section you activate the external reference mode and examine the regenerated signal of interest.

Table 3.14. Settings: generate the signal of interest

Output range Ch1	1 V
Output amplitude Ch1	1.0 V / ON
Frequency Ch1	1 MHz

First demodulator Ch1	ON
Signal input range / AC / Diff / 50 Ch1	1.2 V / ON / OFF / OFF

Use the reference pull-down selector for channel 1 and set it to "Signal Input 2". This uses the signal present at channel 2 to demodulate the signal present at channel 1. Switch to the numerical tool tab.

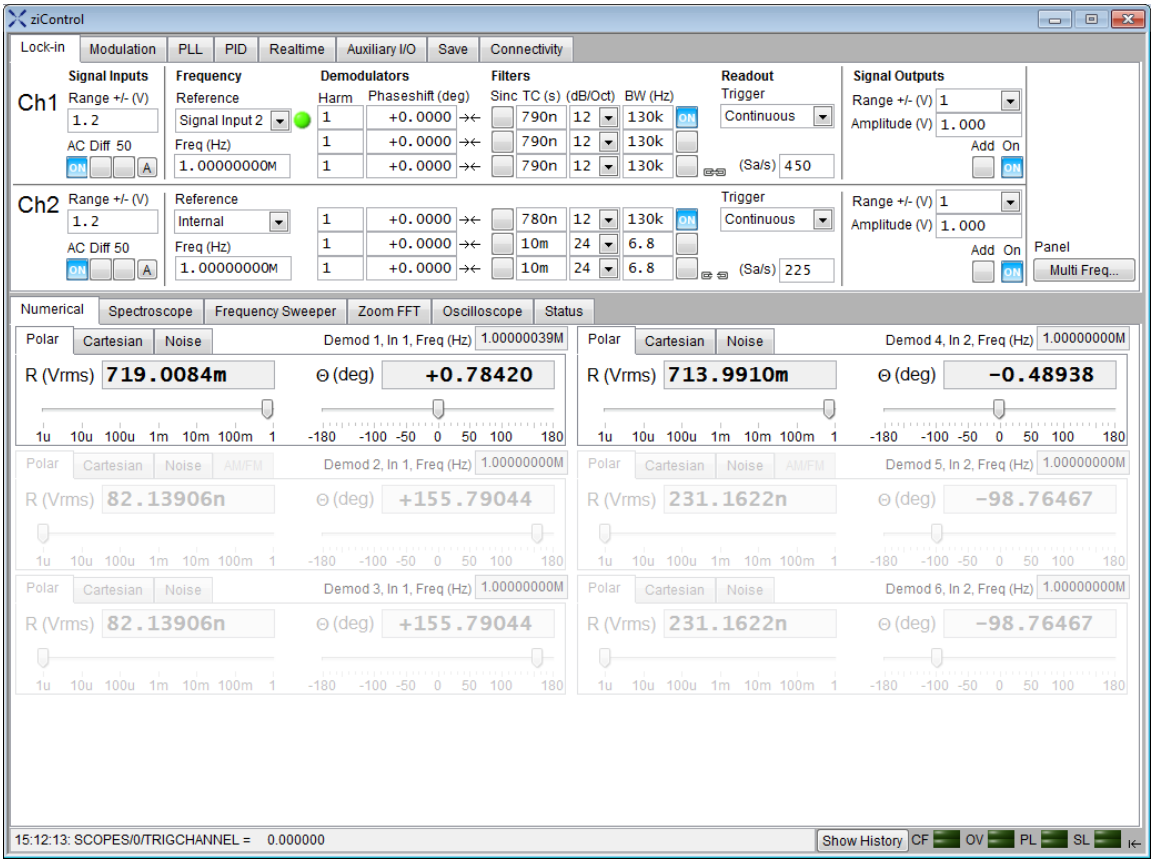


Figure 3.26. ziControl enabling external reference mode

Once the external reference mode has been enabled, the frequency of channel 1 changes continuously, adapting to the frequency of the external reference signal. This can be verified by changing the frequency of channel 2 and noting how the frequency of channel 1 follows. A green indicator appears besides the reference selection for channel 1 indicating that the instrument has locked to an external reference.

At this point, it is worth noting that the external reference signal is not be used directly for demodulation. Instead, the internally regenerated (frequency and phase) reference is used. The continuously changing frequency values express ongoing adaptation of what is regenerated.

The oscillator adapting to the external reference signal can be observed by enabling Output 1. If Output 1 has been connected to Input 1+, the signal can be examined directly using the oscilloscope tool of ziControl with the settings in the following table.

Table 3.15. Settings: acquire the signal of interest

Oscilloscope source	Signal Input 1
Oscilloscope trigger	Signal Input 1
Oscilloscope sampling rate	26 MS, 78 us

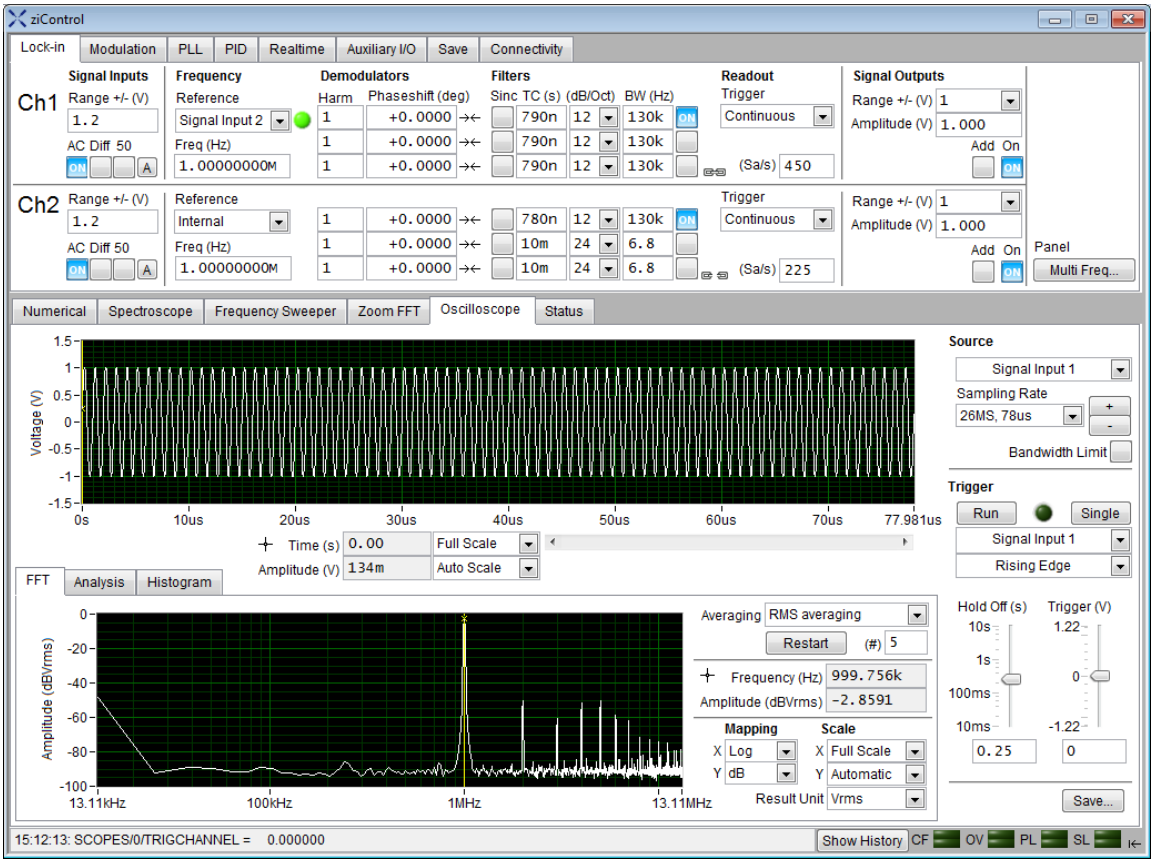


Figure 3.27. ziControl displaying the regenerated signal of interest

3.4.4. Change External Reference Input

In this section you will modify the setup to use DIO 0 as the external reference instead of Input 2. The modified setup is shown on Figure 3.28 . Note that the DIO 0 connector is located on the back panel of the HF2 Instrument.

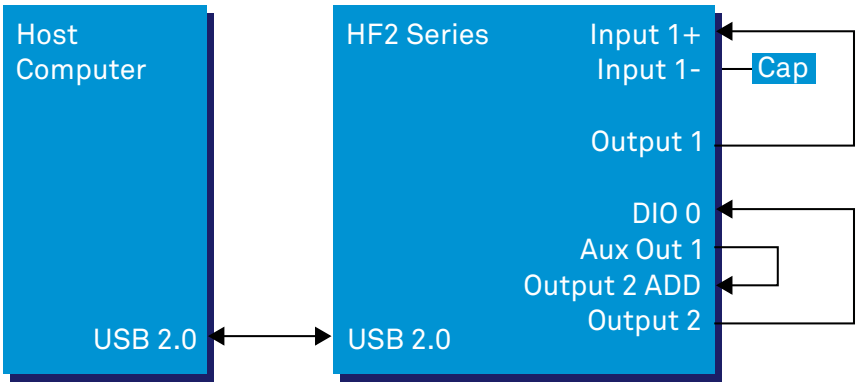


Figure 3.28. Tutorial external reference using DIO 0 setup

It is important to make sure that DIO 0 is configured as an input before connecting anything to it. This can done using the Auxiliary I/O view in ziControl as shown in the following figure. Note that the button to the right of **bits 7...0** should be off.

When using the DIO 0 as the external reference signal, it should be taken into account that this is a digital I/O, which should be operated at TTL levels. Therefore the Aux 1 output is connected to the Add connector of Output 2, to provide a DC shift of the test signal and thus make it TTL compatible.

The settings used for generating the test signal are shown in the following tables. The resulting signal will have a DC offset of 1.5 V and an amplitude of 1 V and will thus oscillate between 0.5 V and 2.5 V, which is TTL compatible.

Table 3.16. Settings: generate the test signal

Output range Ch2	1 V
Output amplitude / ADD Ch2	1.0 V / ON
Frequency Ch2	1 MHz
First demodulator Ch2	ON

Table 3.17. Settings: generate the DC shift

Aux 1 Signal	Manual
Aux 1 Offset	1.5 V

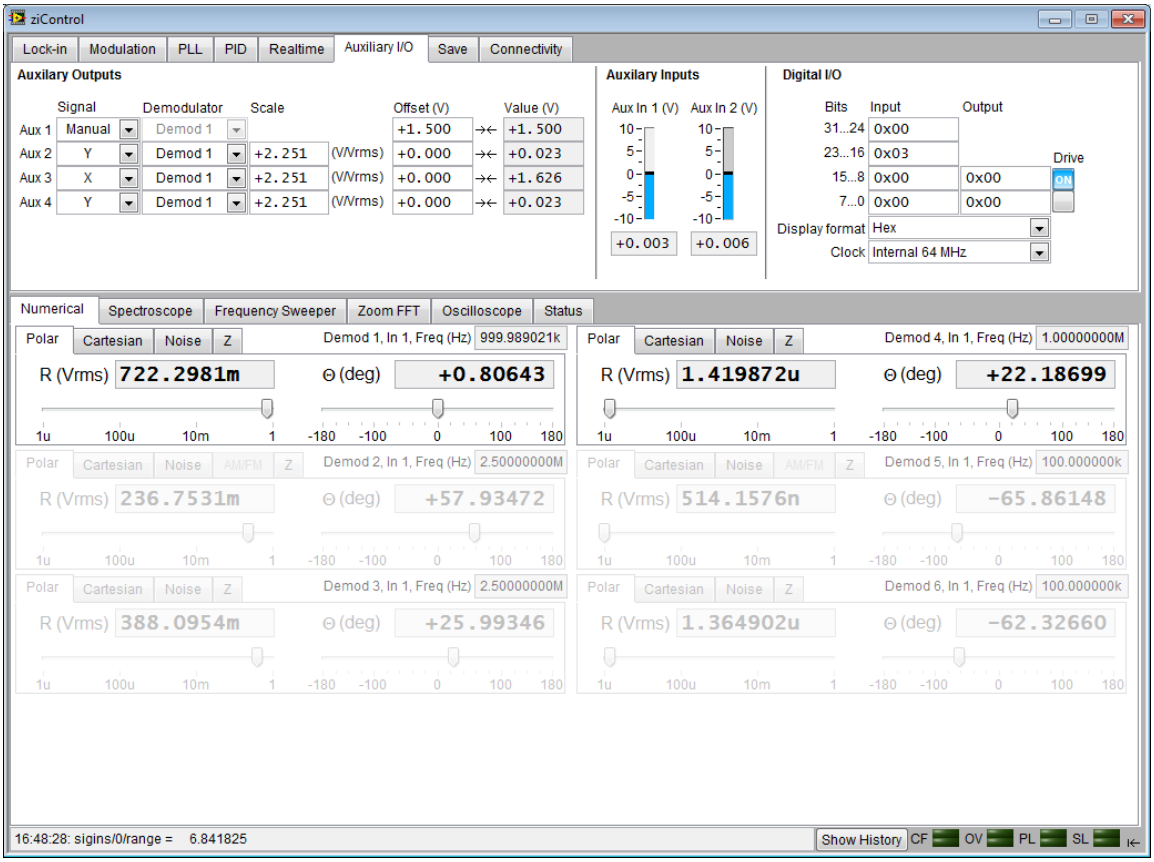


Figure 3.29. Configuring DIO 0 as reference input

Finally use the reference pull-down selector for channel 1 (in the Lock-in tab) and set it to "DIO 0". which makes channel 1 locking on DIO 0 (as shown on Figure 3.30). The frequency of channel 1 should start updating similarly to what was described in Section 3.4.3.

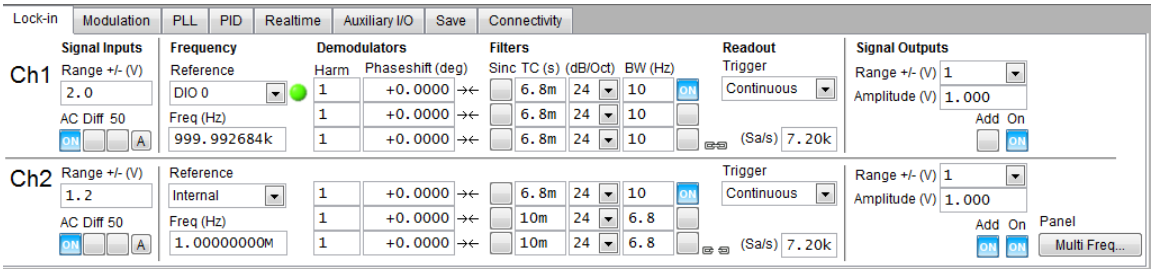


Figure 3.30. Setting DIO 0 as reference for Channel 1

3.5. Tutorial Noise Measurement

Lock-in amplifiers can be used to measure the noise on a signal or a system. The noise property of a system gives detailed information on the maximum achievable performance.

3.5.1. How Does a Lock-in Measure Noise?

A lock-in amplifier measures the signal amplitude close to a given reference frequency with a defined bandwidth around this reference frequency. The noise in an input signal near the reference frequency appears as noise in the lock-in amplifier signal output.

The noise is the standard deviation of the measured X or Y value and is measured by first calculating the average, X_{avg}, over a defined period of time. Then, this signal, X_{avg}, is subtracted from the X value to get the deviation. Finally, the RMS (root-mean-square) is calculated, corresponding to the total noise power of the input signal within a defined bandwidth around the reference frequency. The value is correct for input noise with Gaussian distribution of the noise power, which is normally the case.

Most of the times the noise spectral density is of interest, which is the normalization of the X_{noise} made independent of the filter bandwidth. To calculate the noise spectral density from the calculated RMS noise, one needs to divide the measured value by the square root of the bandwidth √BW. The noise spectral density has the units V/√Hz.

The related equations are X_{noise} = RMS(X - X_{avg}) / √BW, and Y_{noise} = RMS(Y - Y_{avg}) / √BW respectively. The X and Y noise are expected to be identical.

3.5.2. Measuring the Noise of the HF2LI/HF2IS

A LabVIEW example (ziExample-HF2-Noise.vi) is available to measure the noise in an input signal. To measure the equivalent input noise of the HF2, remove all BNC connectors from the input of the device and apply the following settings in the user interface (ziControl).

Table 3.18. Settings: Measure HF2 Noise

Signal input range / AC / Diff / 50 Ch1	0.01 V / ON / OFF / ON
First demodulator Ch1	BW = 100 Hz, dB/Oct = 24
Frequency Ch1	1 MHz
Output switch Ch1	OFF

Run the example, ziExample-HF2-Noise.vi. Make sure, that the correct Demodulator is selected. The noise spectral density should now show a value close to 5 nV/√Hz. By changing the settings in the user interface, the noise behavior of the device can be analyzed in more detail. For example, changing the reference frequency to 10 kHz will slightly increase the spectral noise density, because of flicker noise that is larger at lower frequencies and generally present in all electronic circuits.

3.6. Tutorial Amplitude Modulation

Note

This tutorial only applies to HF2LI lock-in amplifier users that have purchased both HF2LI-MF multi-frequency and HF2LI-MOD AM/FM modulation options.

Amplitude modulation (AM) and frequency modulation (FM) refer to the modulation of an oscillating signal $s(t) = A\cos(\omega t + \varphi)$: A and $\omega t + \varphi$ are respectively the amplitude and phase of the signal. Figure 3.31 depicts the phasor representation of s(t), there the phasor follows the circle with radius A, and the phase wraps around after a full revolution of 360°. The signal s(t) is the projection of the phasor on the abscissa.

In the case of AM signals, the amplitude A, i.e. the phasor length, is time dependent, as in Figure 3.31 (b). In the case of FM signals, the phase φ is time dependent and the phasor has a constant amplitude, see Figure 3.31 (c).

Amplitude and frequency modulations, best known for radio transmission, are also common lock-in detection techniques.

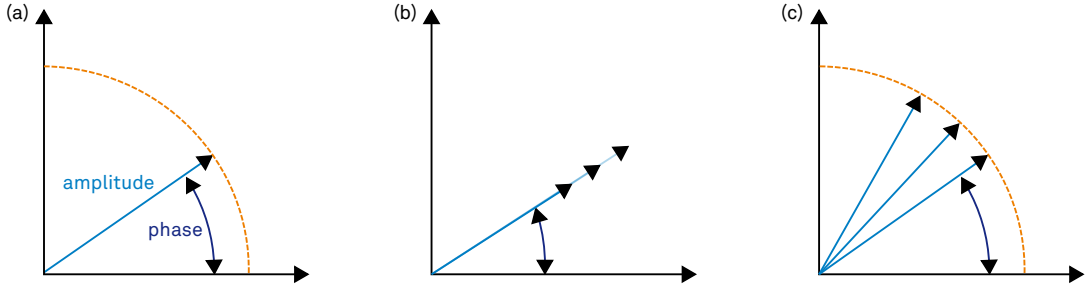


Figure 3.31. A sinusoidal signal represented as a phasor: the signal corresponds to the projection on the x axis. Amplitude (b) and frequency (c) modulated signals affect the amplitude of the phasor or its phase

3.6.1. What is Amplitude Modulation?

In time domain, amplitude modulation of the carrier signal produces a variation of the carrier amplitude proportional to the amplitude of the modulating signal. For example, when the amplitude of a carrier with a frequency $f_c = \omega_c / 2\pi$ is modulated by a signal with frequency $f_m = \omega_m / 2\pi$ (where $f_m < f_c$), the resulting signal has the form

$$s(t) = \left[A + M\sin(\omega_m t)\right]\sin(\omega_c t + \varphi) = A\sin(\omega_c t + \varphi) + \frac{M}{2}\cos\left[(\omega_c - \omega_m)t + \varphi\right] - \frac{M}{2}\cos\left[(\omega_c + \omega_m)t + \varphi\right]$$

Equation 3.1. Amplitude modulation

where A and M are the amplitudes of the fast and slow modulations respectively and φ the phase delay. There is no restriction on the magnitude of M compared to A. The information of interest is encoded in these three parameters, A, M and φ.

In the frequency domain, the AM signal s(t) is composed of three frequencies: the carrier at f_c and two additional sidebands at $f_c - f_m$ and $f_c + f_m$, as shown in Equation 3.1 . The two sidebands

have equal amplitude M/2, half of the modulating signal, and the carrier amplitude is independent on the modulation amplitude.

The traditional way of measuring an AM signal is called double (or tandem) demodulation and requires two lock-in amplifiers: the first one demodulates the signal at f_c with a bandwidth that is at least as large as f_m (but smaller than $f_c - f_m$). This is to ensure that the whole amplitude of the modulation signal is retained. The demodulated signal after the first lock-in becomes

$$s(t) \cdot \cos(\omega_c t) \xrightarrow{\text{after filtering}} d_1(t) = \frac{A + M \sin(\omega_m t)}{2} \cos\varphi$$

Equation 3.2. Tandem demodulation

In $d_1(t)$, the two sidebands are now located at the same frequency f_m, while the carrier appears as a DC component. When the demodulated signal $d_1(t)$ is fed to a second lock-in, the result of the second demodulation at f_m is proportional to M.

To recover the phase φ, one could make use of a third lock-in to demodulated s(t) with a bandwidth smaller than f_m as shown in see Figure 3.32 (note that one must recover the phase because this enters as a systematic error in the modulation amplitude as $M \cos\varphi$.

While the HF2LI can be used in tandem configuration, the HF2LI-MOD option allows the user to demodulate directly at the three frequencies f_c and $f_c \pm f_m$ simultaneously, using one lock-in unit only and reducing the complexity of the setup. Therefore, selecting a filter with bandwidth smaller than f_m, the three parameters A, M and φ are directly measured and displayed.

Internally, the HF2LI generates the phases $\omega_c t$ and $\omega_m t$ from which it produces $(\omega_c - \omega_m)t$, $\omega_c t$ and $(\omega_c + \omega_m)t$: this ensures the correct phase relationship for the demodulations of the sidebands.

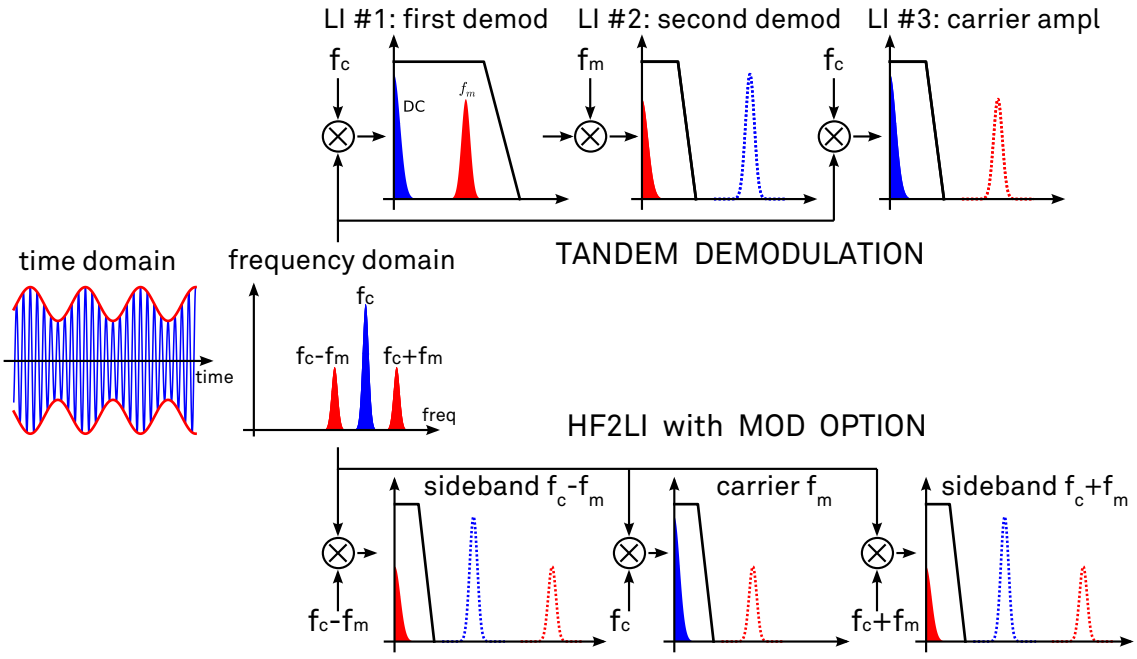


Figure 3.32. Comparison between tandem demodulation and the HF2LI-MOD option of an AM modulated signal

3.6.2. Generate the Test Signal

In this tutorial, you are going to generate an AM signal with a carrier frequency of 1 MHz and a modulation frequency of 100 kHz. The signal is generated at Signal Output 2 and is demodulated by the first lock-in unit by feeding it into Signal Input 1. The HF2LI-MOD option requires the HF2LI-MF option to generate and demodulate AM or FM signals: changing the settings of Mod1 and Mod2 in ziControl Modulation tab will modify some of the settings typically in the Lock-in MF tab. The reader is referred to Section 4.2.3 .

In the Modulation tab, in the Mod2 section, select the following parameters:

Table 3.19. Settings: generate the AM signal

Carrier frequency / Frequency	Osc 1 / 1 MHz
Signal Frequency / Frequency	Osc 2 / 100 kHz
Mode	AM
Sideband	Both
Carrier / Modulation	200 mV / 100 mV

This generates an AM signal with the two sidebands of equal amplitude. To look at this signal, connect Signal Output 2 to Signal Input 1 of the HF2LI.

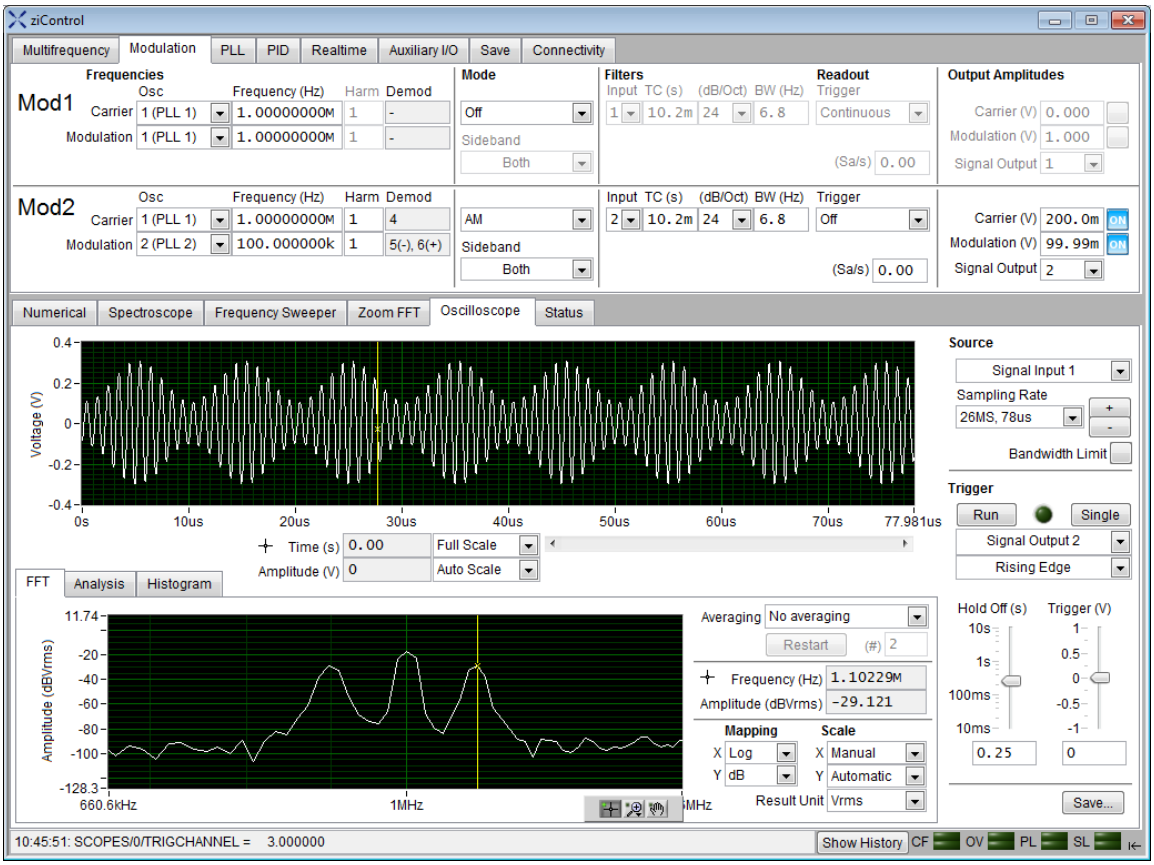


Figure 3.33. The AM signal generated by Mod2 as measured on Signal Input 1

Select the correct input parameters: in the Lock-in tab, for the Ch1 section, make sure Differential mode and 50 Ω are turned off. Then click on the auto-range button (marked by A). In the oscilloscope tab, select Source to be Signal Input 1, Trigger to be Signal Output 2 and click on Run

to activate the oscilloscope. Observe how the carrier amplitude is modulated at 100 kHz. The FFT tab underneath shows three peaks: the carrier frequency at 1 MHz and two sidebands at 0.9 and 1.1 MHz (see the yellow cursor in the FFT tab of Figure 3.33).

3.6.3. Measure the Test Signal

In the Modulation tab, in the Mod1 section, select the following parameters:

Table 3.20. Settings: measure the AM signal

Carrier frequency	Osc 1
Signal Frequency	Osc 2
Mode	AM
Sideband	Both
Filter BW	10 Hz
Readout	Continuous

This sets the correct demodulation of the AM signal with the two sidebands. In the Numerical tab, look at the amplitude of the carrier, 142 mV_{RMS} and of the two sidebands, 35 mV_{RMS} each, one quarter of the carrier amplitude: this corresponds to a modulation index of 50%. Indeed, on the Numerical indicator for Demodulator 2, click on the AM/FM tab: there you can read directly the measured modulation index h and the measured modulation index M.

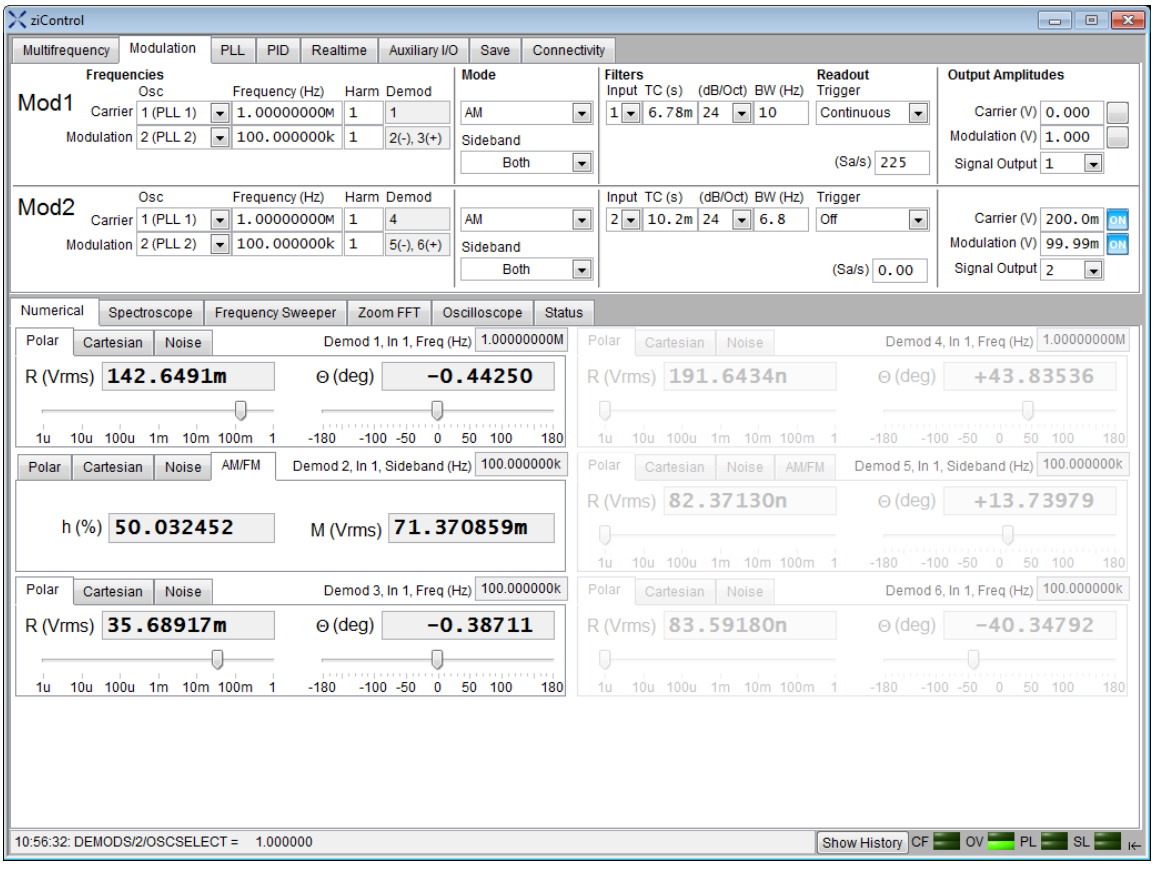


Figure 3.34. The numerical view of the demodulated signal

The HF2 directly demodulates simultaneously on each one of the three frequencies, therefore using three demodulators: this appears in the Numerical tab, where 3 indicators are running, or in the multi-frequency tab, where three readout outputs are enabled.

3.7. Tutorial Frequency Modulation

Note

This tutorial only applies to HF2LI lock-in amplifier users that have purchased both HF2LI-MF multi-frequency and HF2LI-MOD AM/FM modulation options.

3.7.1. What is Frequency Modulation?

In frequency modulation (FM), the amplitude of the modulating signal is proportional to the instantaneous frequency deviation from a fixed frequency. Thus in the simplest case, the FM signal of Figure 3.35 (a)

$$\mathrm{s}(t) = A\cos\left[\omega_c t + \frac{\omega_p}{\omega_m}\sin\left(\omega_m t\right) + \varphi\right]$$

Equation 3.3. Frequency modulation

is produced when a carrier signal of frequency $f_c = \omega_c / 2\pi$ is modulated by a sinusoidal modulation with frequency $f_m = \omega_m / 2\pi$. The maximum variation of the frequency around the carrier frequency, the peak frequency deviation is $f_p = \omega_p / 2\pi$: in fact, because the frequency is the time derivative of the phase (divided by 2π) and the phase is the argument of the cosine in equation Equation 3.3 , the instantaneous frequency is

$$\mathrm{f}(t) = f_c + f_p\cos\left(2\pi f_m t\right)$$

Equation 3.4. Signal instantaneous frequency

Here, the physical information is encoded in the parameters A, f_p and φ.

The spectrum of the FM signal of Equation 3.3 is mathematically complex: it consists of the carrier and a series of pairs of sidebands at multiple integers of f_m around the carrier frequency, see Figure 3.35 (d). The amplitudes of the carrier and sidebands are given by mathematical functions called Bessel functions usually indicated by J_n) evaluated at the modulation index $h = f_p / f_m$. For instance, the n-th pair of sidebands is located symmetrically around f_m at $f_c \pm n f_m$ and its amplitude is $J_n(h)$.

A peculiarity of the Bessel functions is that they oscillate around zero: even for the carrier, as the modulation index is increased, its amplitude $J_0(h)$ decreases, crossing zero at h ≈ 2.41 and then it increases in amplitude (in anti-phase) before reaching zero again at h ≈ 5.52.

At low modulation indexes, the amplitude of higher sidebands is very low and can thus be ignored: this is called the narrow band approximation. In this limit (it is customary to assume h < 0.2), only the two sidebands at $f_c \pm f_m$ have non-negligible amplitude and the signal s(t) can be approximated by

$$\widetilde{\mathrm{s}}(t) = \mathrm{A}\left[J_0(h)\sin\left(\omega_c t + \varphi\right) - J_1(h)\cos\left[(\omega_c + \omega_m)t + \varphi\right] + J_1(h)\cos\left[(\omega_c - \omega_m)t + \varphi\right]\right]$$

Equation 3.5. Approximated narrow band FM signal

The first term is the carrier, the other two are the lower and upper sidebands. The problem of finding h (and the peak amplitude f_p) reduces now to comparing the amplitude of the first pair of sidebands and the carrier to the ratio $J_1(h)/J_0(h)$. A plot of the ratio $J_1(h)/J_0(h)$ and $J_2(h)/J_0(h)$ is in Figure 3.35 (e).

Even though $\tilde{s}(t)$ looks very similar to an AM signal, there is however a substantial difference in that the phase of the lower sideband is reversed: the difference between an AM and a narrow band FM signal is indeed subtle! This results in the sidebands being in quadrature with the carrier. For example, assume that φ = 0: demodulating $\tilde{s}(t)$ with the carrier signal $\sin(\omega_c t)$ gives the DC component (the carrier) but no sidebands; on the other hand, demodulating with the quadrature $\cos(\omega_c t)$, only the two sidebands at f_m are observed and no carrier is present. Because of this, FM detection can be done in a similar way as the AM demodulation scheme, using the tandem configuration described previously in Section 3.6.1 .

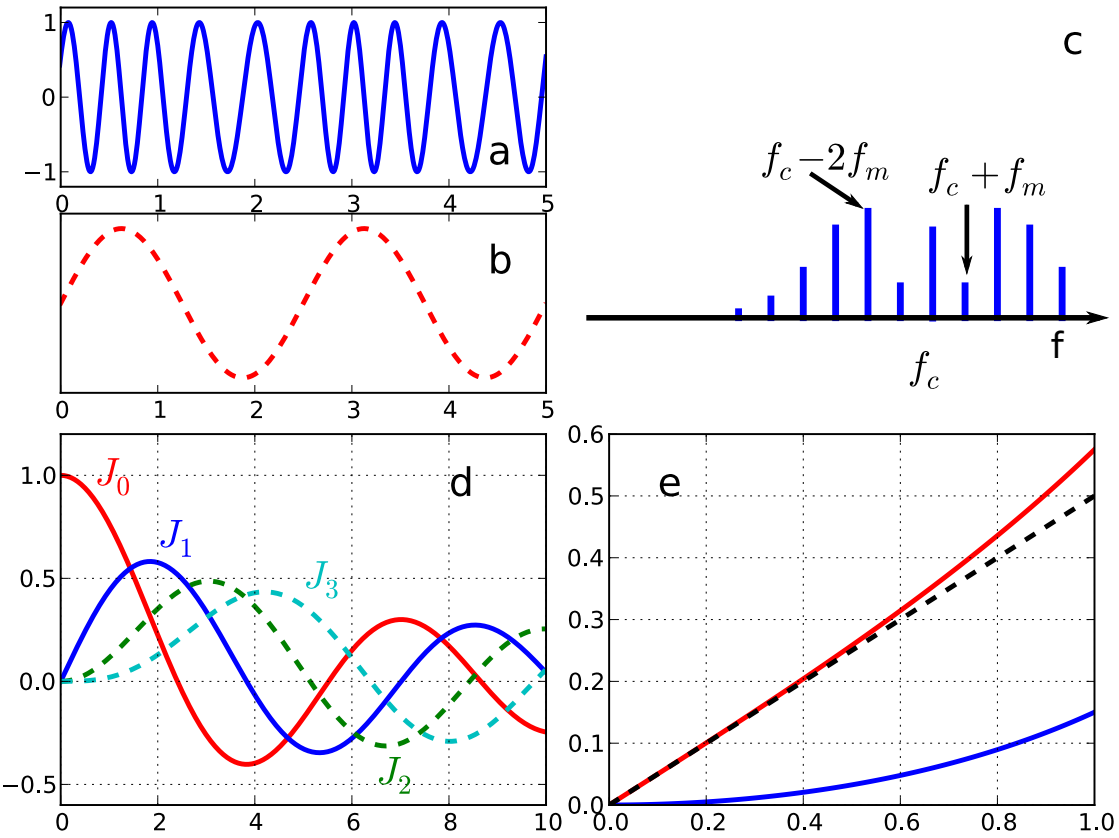


Figure 3.35. (a) A simple frequency modulated signal, (b) its instantaneous frequency, (c) the frequency domain spectrum of a FM signal is composed of an infinite series of sidebands, here depicted for h = 0.35, (d) n-th Bessel function versus h, (e) ratio $J_1(h)/J_0(h)$ (red line), $J_2(h)/J_0(h)$(blue line), slope 0.5 line (black line)

The HF2LI-MOD modulation option permits direct generation/demodulation of a FM signal. For demodulation, this option measures and displays the parameters A, f_p and φ.

Internally the HF2LI calculates the peak frequency f_p with the method described above, from the ratio $J_1(h)/J_0(h)$, proportional to the carrier and first sideband amplitude. Therefore this

method is expected to give poor accuracy when the carrier amplitude is small, at h = 2.1 (at values much larger than narrow band frequency modulation for which h < 0.2): for the best performance, users are recommended to limit the demodulation to values of the modulation index h < 1, where $J_1(h)/J_0(h)$ has the largest sensitivity.

Users interested in demodulating an FM signal are advised to use the Modulation option instead of using the PLL to track the frequency deviation Δf and to further demodulate this signal. The first approach guarantees a better signal to noise ratio because the signal can be averaged over several modulation cycles while keeping the bandwidth small.

3.7.2. Generate the Test Signal

In this tutorial, you are going to generate an FM signal with a carrier frequency 1 MHz and a modulation frequency of 100 kHz and a peak frequency 50 kHz. The signal is generated at Signal Output 2 and is demodulated by the other lock-in unit by feeding it into Signal Input 1.

In the Modulation tab, in the Mod2 section, select the following parameters:

Table 3.21. Settings: generate the FM signal

Carrier frequency / Frequency	Osc 1 / 1 MHz
Signal Frequency / Frequency	Osc 2 / 100 kHz
Mode (Mod1)	FM Demodulator
Sideband	Both
Mode (Mod2)	FM Generator
Carrier / Peak Frequency	100 mV / 50 kHz

This generates an FM signal and several sidebands, the largest ones at $f_c \pm f_m$. To look at this signal, connect Signal Output 2 to Signal Input 1 of the HF2 Instrument.

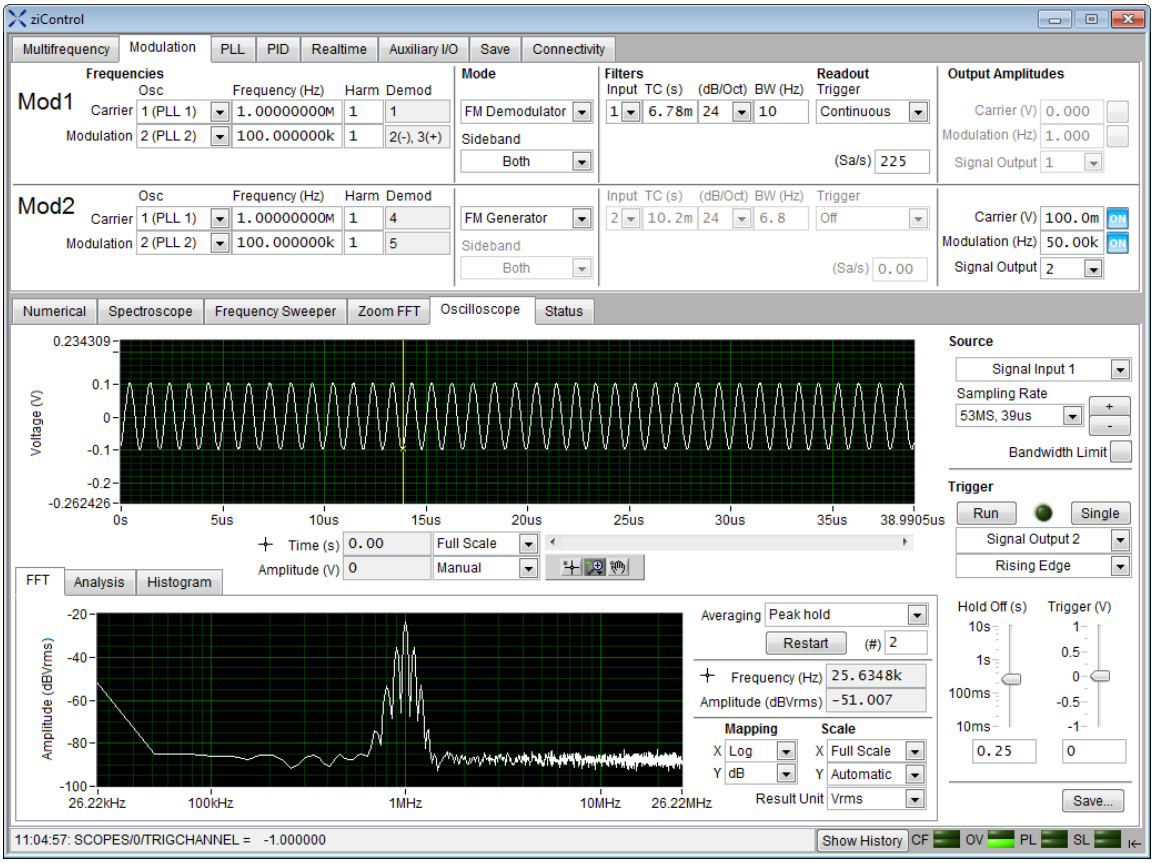


Figure 3.36. The FM signal generated by Mod2 as measured on Signal Input 1

Select the correct input parameters: in the Lock-in tab, for the Ch1 section, make sure Differential and 50 Ω are turned off. Then click the auto-range button (marked by A). In the oscilloscope tab, select Source to be Signal Input 1, Trigger to be Signal Output 2 and click on Run to activate the oscilloscope. Observe that the carrier amplitude is constant but the distance between two successive peaks of the sinusoidal wave changes periodically. The FFT tab underneath shows the carrier at 1 MHz and several sidebands with decreasing amplitude as the sideband order increases: you can increase the resolution by selecting a larger Time scale in the Source section of the Oscilloscope.

3.7.3. Measure the Test Signal

In the Modulation tab, in the Mod1 section, select the following parameters:

Table 3.22. Measure the FM signal

Carrier frequency	Osc 1
Signal Frequency	Osc 2
Mode	FM Demodulator
Sideband	Both
Filters BW	10 Hz

Readout	Continuous

This sets the correct demodulation of the FM signal. In the Numerical tab, look at the amplitude of the carrier, 67.2 mV_{RMS}: it is not a surprise that we do not read 71 mV_{RMS} because at h = 50 kHz / 100 kHz = 0.5, the value of the Bessel function $J_0(0.5)$ is smaller than 1. You can also see that the two sidebands have an amplitude of 17 mV_{RMS} each but their phases are 180° apart. On the Numerical indicator for Demodulator 2, click on the AM/FM tab: the peak frequency is correctly determined to be 50 kHz.

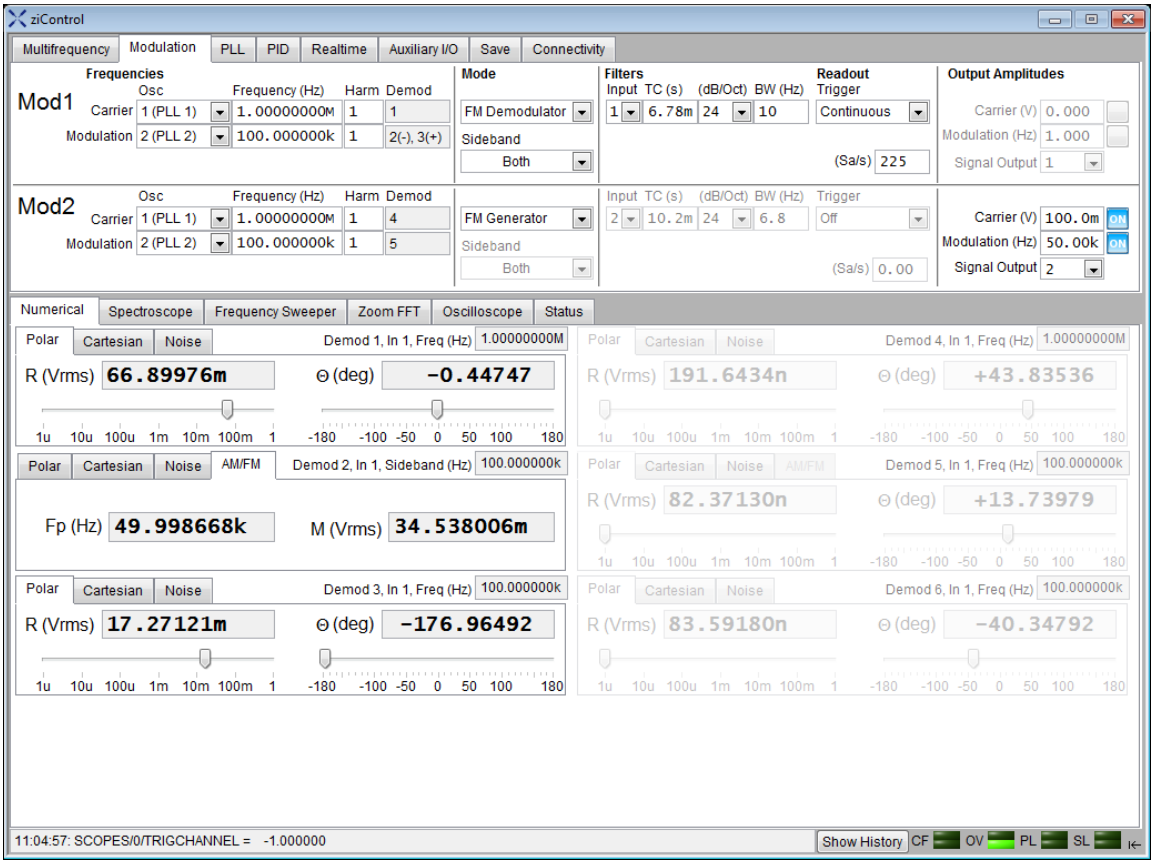


Figure 3.37. The numerical view of the demodulated signal

The HF2 Instrument directly demodulates simultaneously on each one of the three frequencies, therefore using three demodulators: this appears in the Numerical tab, where 3 indicators are running, or in the multi-frequency tab, where three readout outputs are enabled.

Optionally, you can check what is the maximum modulation index you can reliably demodulate for your specific application: in the Mod2 section, gradually increase the peak frequency to 240 kHz, for which carrier amplitude reaches 0. Indeed theoretically at h ≈ 2.41 the Bessel function J_0 reaches zero: for this reason we recommend to limit the modulation index to less than 1.0.

3.7.4. Notes about FM Generation

- The HF2LI-MOD option requires the HF2LI-MF option to generate and demodulate AM or FM signals: changing the settings of Mod1 and Mod2 in ziControl Modulation tab will modify

some of the settings typically in the Lock-in MF tab. The consistency of such operation needs to be checked after setting modification. For more information on resource use, the reader is referred to Section 4.2.3 .

- As explained above, for FM generation, one demodulator only is used and its instantaneous frequency changes according to Equation 3.4 . Care should be exercised when using such an FM generating demodulator for the demodulation of other signals. Consider for instance the following situation, in which a 1 MHz 100 mV sine wave is generated using demodulator 7 (connected to oscillator 1) and its spectrum is detected by the zoomFFT using demodulator 3, also attached to oscillator 1: one single peak at 0 Hz relative frequency is visible, as expected, since we are demodulating at the same frequency as the input signal.

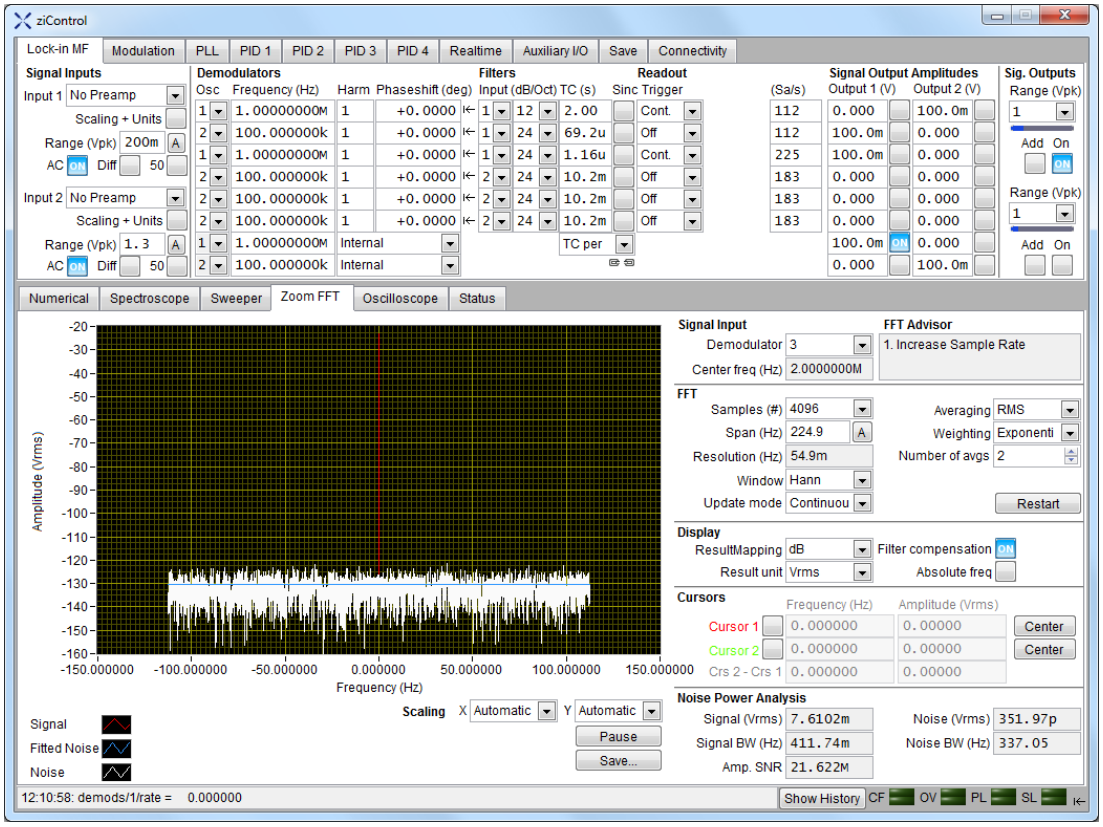


Figure 3.38. Demodulation of a 1 MHz input signal at 1 MHz

Now turn on FM generation on Mod1 using oscillator 1 for the carrier, oscillator 6 for the modulation frequency with $f_m = 100\text{Hz}$, and $f_p = 200\text{Hz}$ (Modulation index (f_p/f_m)= 2): this will make demodulator 1 instantaneous frequency time dependent.

When demodulation is done using demodulator 1, the frequency spectrum in the zoomFFT resembles the typical FM spectrum (since this is not a narrow band FM signal (h>0.2), several sidebands are visible). This can be understood using the rotating frame picture: when a frequency modulated observer is trying to detect a steady signal, it appears in his reference frame as being frequency modulated. This is like the drunken person who is perceiving the whole world sloshing around him.

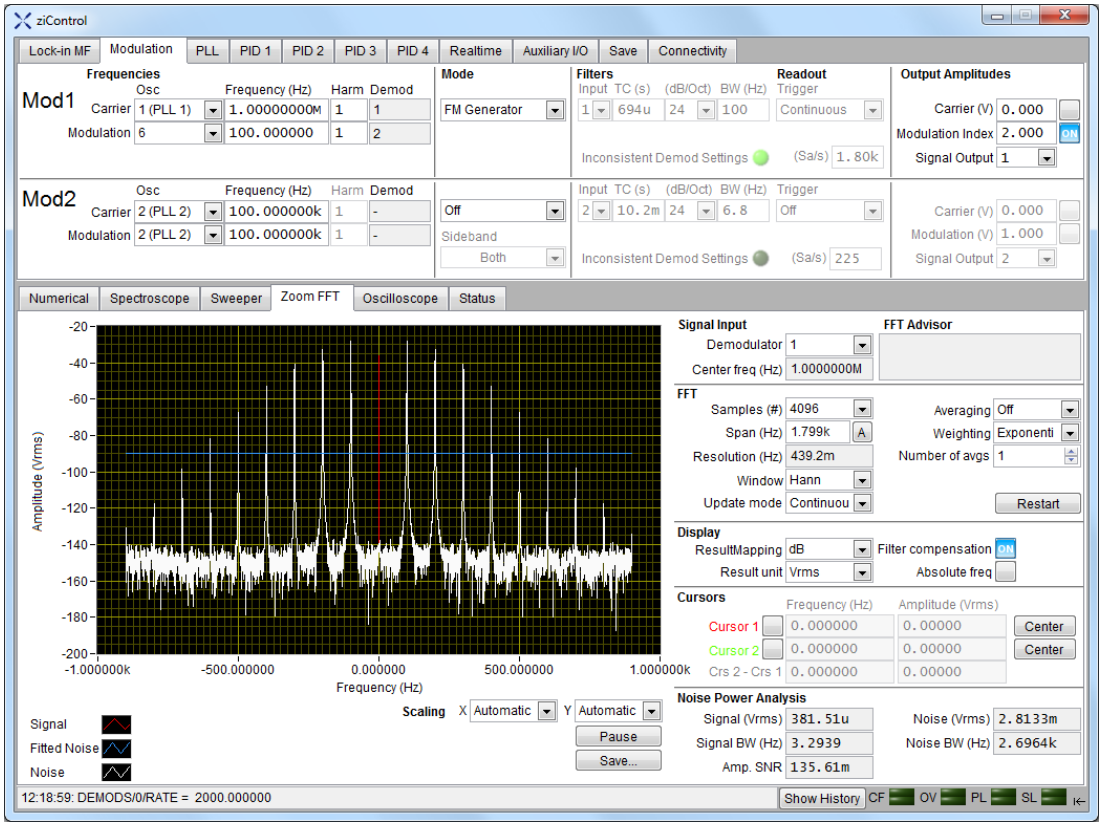


Figure 3.39. Demodulation of a 1 MHz input signal with an FM generating demodulator

3.8. Tutorial Phase Locked Loop

Note

This tutorial only applies to HF2LI users that have purchased the HF2LI-PLL Phase-locked Loop option.

3.8.1. What is a Phase Locked Loop?

The phase-locked loop (PLL) is a control system that produces a reference signal whose phase is related to that of the input signal. A common use of the PLL is to excite a resonating system: for such systems, there is a phase delay between the drive signal (produced by the PLL) and the system response which at resonance, is typically 90°. A change of the resonance frequency (for example as a result of the interaction between tip and sample as in AFM systems) induces a change of the phase delay between the drive and the response. The PLL is designed to adjust the drive signal frequency in such a way to restore the required phase delay. The maximum rate at which the PLL can restore the correct phase delay is called the bandwidth of the PLL and is one of the figures of merit of the PLL.

A PLL consists of three parts arranged in a feedback configuration: a phase detector (PD), a proportional-integral (PI) controller, and a variable frequency oscillator. The phase detector compares the phase of the input signal to that of the reference signal derived from the oscillator. Their phase difference must be made equal to the phase set point: any deviation from the set point is the error signal and has to be minimized. This is the role of the PI controller and the oscillator. If the error is positive, the reference signal is lagging behind the rate at which the input signal is advancing and the oscillator frequency has to increase. If the error is negative, the reference signal is ahead and the oscillator frequency has to be decreased. The name phase locked loop reflects the fact that the two signals are related by their phases. As a consequence of this, the reference signal has the same frequency as the input signal because the frequency is the derivative of the phase.

Defining Θ being the measured phase error between reference and input signal, f_c the center frequency which is typically set to a value close to the operating frequency of the PLL, T_i the integral gain and K_p the proportional gain of the PI controller, then the reference frequency generated by the variable oscillator satisfies

$$f_r = f_c + k_p\left[\Theta + \frac{1}{T_i}\int_t \Theta \mathrm{dt}'\right]$$

Equation 3.6. PLL generated reference frequency

The purpose of the PLL is to accurately track frequency changes. For instance in the case of a FM input signal (see Section 3.7.1), in which the instantaneous frequency varies as $\mathrm{f}(t) = f_c + f_p \sin\left(2\pi f_m t\right)$, one would like to extract f_p. If the modulation frequency f_m is too large, the PLL will not be able to track the frequency change and f_p will appear smaller than it actually is. The bandwidth of the PLL is therefore a relevant performance parameter and is defined as the frequency f_m at which the signal of interest f_p is measured to be attenuated by 3 dB compared to the original signal. When correctly set up, the PLL transfer function should be as flat as possible before rolling off at the 3 dB point. On the other hand, an unnecessarily wide PLL bandwidth will reject less noise.

Being a feedback control system with gain (given by the PI), the PLL is susceptible to instability if the parameters are not chosen correctly. The scientific literature abounds with methods describing different more or less complicated procedures to tune the PI (and the PLL).

3.8.2. Generate the Test Signal

In this tutorial, you are going to learn how to set up the PLL parameters in relation to the bandwidth of the signal you want to observe. To do this, you are going to generate an FM signal and make use of the PLL Advisor to find out the correct parameters.

HF2LI users that have the HF2LI-MOD option installed on their HF2 Instrument should select the following parameters for the Mod2 unit:

Table 3.23. Settings: generate the FM signal for PLL tracking

Carrier oscillator / frequency	Osc 5 / 1 MHz
Signal oscillator / frequency	Osc 6 / 20 kHz
Mode	FM Generator
Sideband	Both
Carrier amplitude / Peak frequency	100 mV / 2 kHz
Output select	Signal output 2

Connect a BNC cable between Output 2 and Input 1+ and turn Output signal On. You may need to check in the Lock-in MF tab that only demodulator 4 is enabled (typically you would need to turn off demodulator 8).

HF2LI users without the HF2LI-MOD option installed could use an external frequency generator to produce an FM signal with carrier frequency of 1 MHz, modulation frequency of 20 kHz and peak frequency of 2 kHz. Connect the output of the frequency generator to the HF2LI Input 1+.

3.8.3. Measure the Test Signal

The objective is to accurately measure accurately the frequency modulation with suitable PLL settings.

The PLL is a feedback system that has four adjustable parameters: the time constant and the filter order of the phase detector and the proportional gain and time constant of the PI controller. As a rule of thumb, the bandwidth of the phase detector could to be set to about twice the target PLL overall bandwidth: if the PD would be limiting the PLL bandwidth then the PLL transfer function would show overshoots and make the PLL unstable. On the other hand, a too large bandwidth on the PD would transfer more noise into the PI.

Apply the following settings to one of the PLLs:

Table 3.24. Settings: prepare the PLL for FM signal tracking

Center frequency / Frequency range	1 MHz / 100 kHz
Phase detector and PI controller settings	Auto Full Bandwidth

In the Auxiliary I/O tab set dF as the analog output for Aux 1, set the scale to 100 µV/Hz (based on the expected maximum frequency swing) and connect a BNC cable from Aux 1 to Signal Input 2 to measure the frequency deviation dF of the signal on Input 1: we expect to observe a signal with

amplitude of 2 kHz*100 µV/Hz = 200 mV. Under the Lock-in tab, set the following parameters: AC coupling, 50 Ω disabled, Diff disabled, 300 mV input range; under the Oscilloscope tab, trigger on Oscillator 5 for users using the HF2LI-MOD option (or Continuous if using an external generator), and select Input signal 2, Sampling Rate 1.6 MS, 1.2 ms.

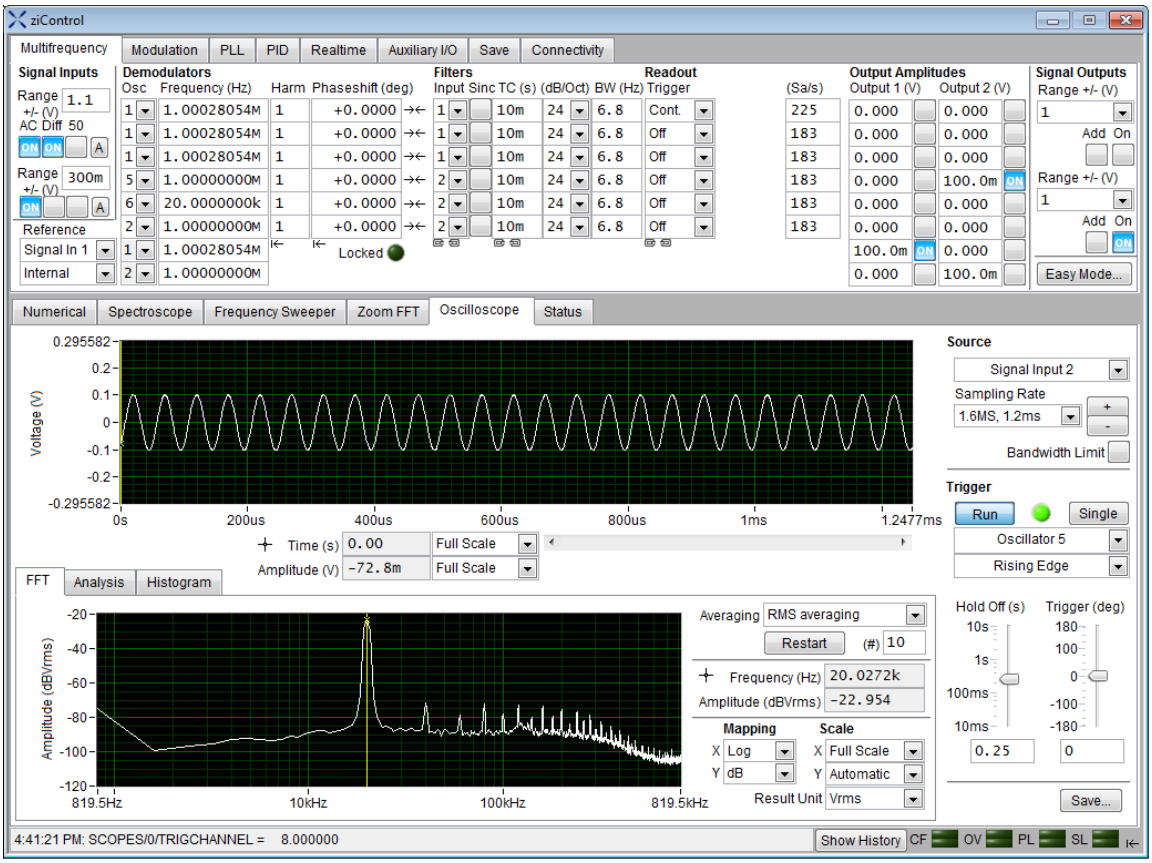


Figure 3.40. The HF2PLL dF output signal with gain of 100 V/Hz

Click on Run and observe the signal. Its amplitude is approximately 100 mV, which is only 50% only of the expected value for the depth frequency (you may also want to set PLL/0/AUXAVG to 1 using the text-based programming as described in Section 6.2.2 to achieve a smooth signal, although this is not absolutely necessary). We must conclude that the PLL bandwidth is smaller than the modulation frequency, 20 kHz.

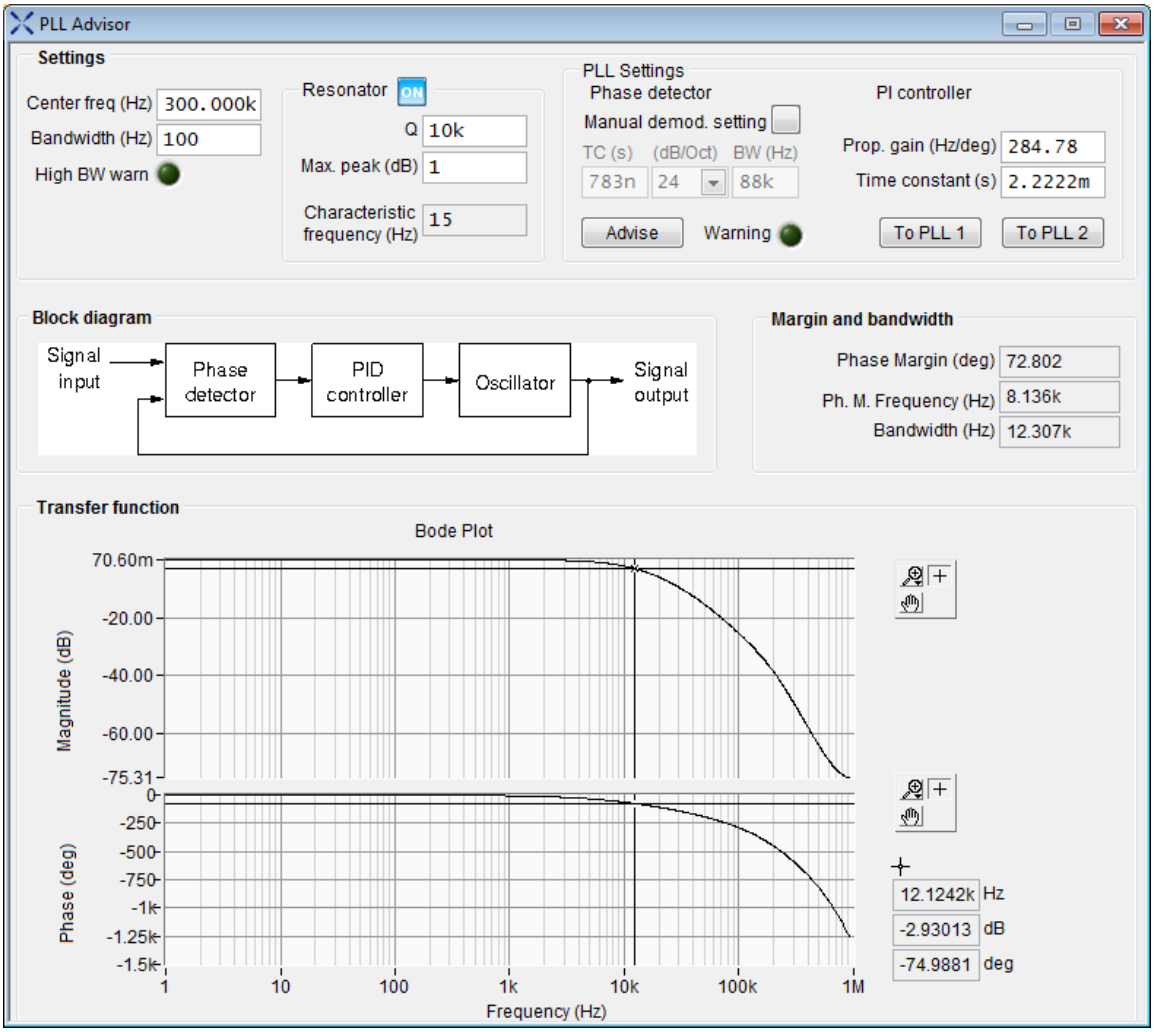


Figure 3.41. The HF2PLL Advisor showing the transfer function of the PLL for a given set of parameters

We are now going to find out the right parameters. Enable the PD filters settings and the PI settings and click on Advisor. You will be presented with a panel displaying the simulated PLL transfer function: you can change the parameters on the Advisor panel, look at the response and once you are satisfied, apply them to PLL1 or PLL2. The Advisor contains an algorithm with a pretty accurate model of the HF2LI PLL to let you predict with a high degree of confidence the PLL response.

On the Advisor, input the PLL current values for the time constant, the filter order, the proportional gain and the integration time (the values you will read from PLL1 in the PLL tab). Read the (simulated) PLL bandwidth by looking at the 3 dB point in the transfer function Bode plot, 11.7 kHz. (Alternatively, you can read the value of the 3 dB point in the numerical display at the right hand side). At 20 kHz, the attenuation is -6.17dB, around 49%: this is very close to the experimentally observed value of 100 mV.

With these settings, the phase detector filter bandwidth is 88 kHz, exceeding the factor of 2 the rule mentioned before. Increase the time constant until the BW is 41 kHz (TC is now 1.7 µs). Now increase the proportional gain, until the BW becomes approximately 20 kHz: with K_p at 350 we obtain a bandwidth of 20 kHz. On the Advisor, click the To PLL1 button: the current parameters are transferred to the PLL tab. On the oscilloscope, look at the new amplitude of the signal at Signal Input 2, about 150 mV, because 20 kHz is the PLL 3 dB point.

The Advisor closely reflects the response of the PLL. Figure 3.42 is a comparison between the Advisor (solid blue line) and the PLL (red dots) to a FM signal that is fed to the PLL input, using the parameters above: see how in the whole PLL range of interest, the difference between the two traces is less than 0.2 dB.

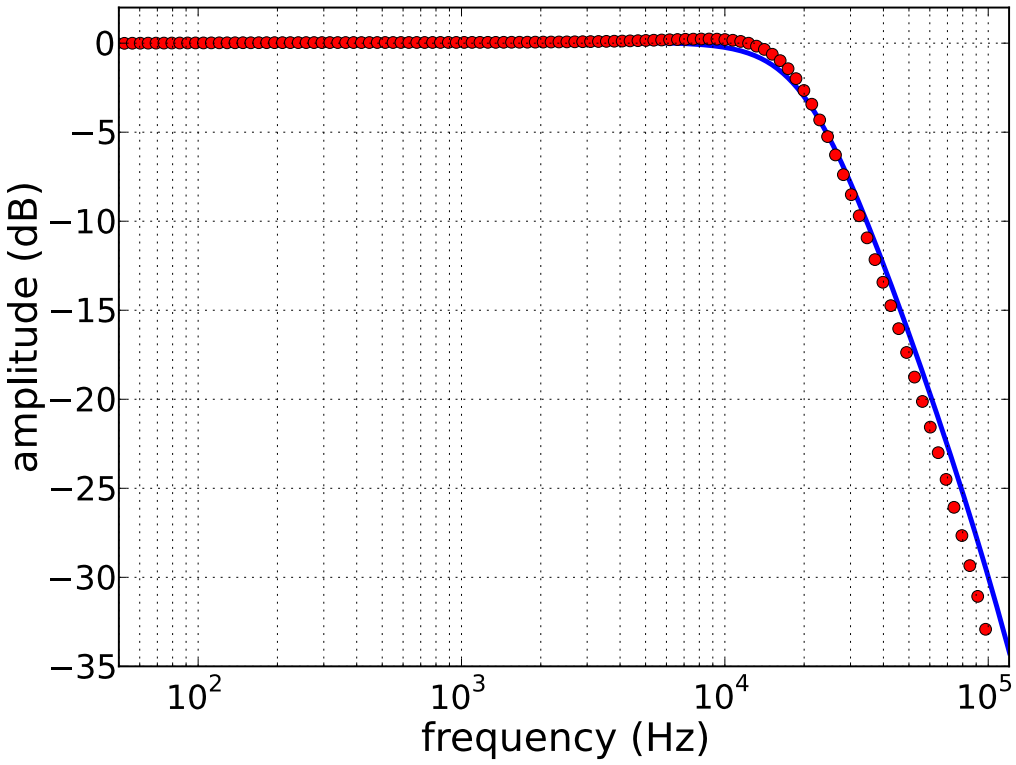


Figure 3.42. Comparison between the Advisor (blue line) and PLL response (red dots) to a FM signal: this confirms that the Advisor is an excellent model of the PLL and can be relied on to set up the correct parameters

What happens if the proportional gain is too high? Try to change it to 600. The transfer function presents now an overshoot of 4.3dB at 20.5 kHz. Transfer these parameters into PLL and measure the amplitude of dF, 300 mV, larger than the expected value of 200 mV by about 3.5 dB. In general a larger ratio K_p/T_i makes the PLL respond faster (i.e. it increases the bandwidth).

3.9. Tutorial PLL/Resonator

One of the typical uses of the PLL is to drive a resonator at its resonance frequency, which may change as a result of perturbation by external forces (e.g. mass changes in quartz crystal micro balance measurements, or surface forces in AFM measurements). As mentioned in Section 3.8.1 , the PLL ensures a fixed phase difference between its input and the output signals (e.g. -90°), independently of the signal instantaneous frequency (within the PLL bandwidth). It also output the frequency shift from a center frequency which physically represents the amount of perturbation on the resonator.

This tutorial shows how to set up the PLL on a quartz resonator. In the example, a commercial quartz crystal with resonance frequency of approximately 16 MHz is shorted by a large resistor (to decrease the quartz large Q factor) and connected between Signal Output 1 and Signal Input 1; set the excitation voltage to 100 mV peak amplitude. We first need to find the resonance frequency: begin with a frequency sweep with wide range 15 MHz and 17 MHz, 300 points (it does not matter whether the sweep is logarithmic or linear or whether Auto BW is enabled). The sweep will show several resonances, each one consisting of an impedance maximum and minimum (this is the result of the way the quartz is usually cut). Zoom in the region of the resonances, using the magnification tool and the button Copy from Plot Range (more than once if necessary) and drag the yellow cursor on the first largest resonance: this resonator has a resonance frequency at 15.997 MHz, where the phase is -100°. One should not be surprised by this because at higher frequencies, the cable length significantly delays the signal propagation.

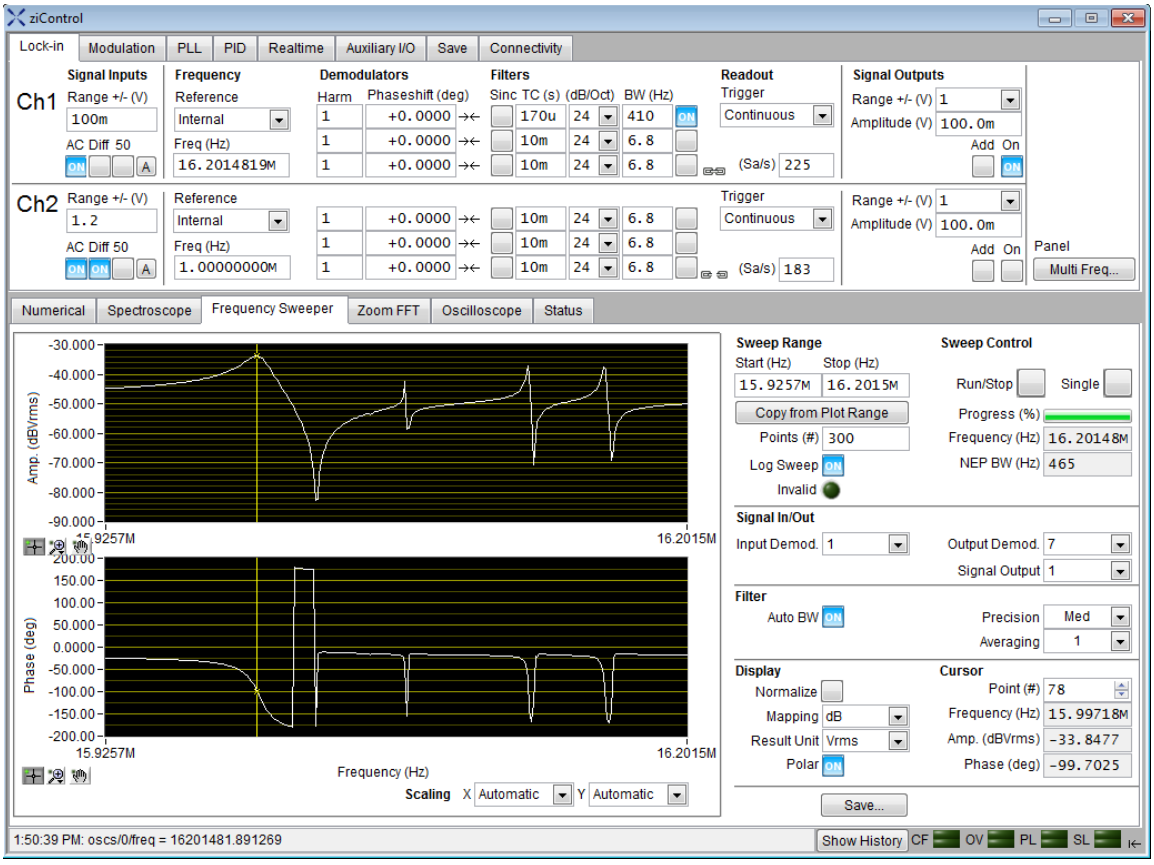


Figure 3.43. Frequency sweep of the quartz resonator (zoom in of the four resonances)

To to the PLL tab, copy the center frequency and the phase setpoint (15.997 MHz and -100°) into the corresponding boxes of PLL1 and reduce the range to 100 kHz. To find an initial set of

parameters for the phase detector and PID controller, use the Advisor. Let us assume that we need a bandwidth of 1 kHz (this is just an academic exercise as the quartz is cased and its resonance frequency will not change so fast): type the approximate resonance frequency and the bandwidth (16 MHz and 1 kHz) into the boxes on the top left side. Disable auto full bandwidth and type 3 kHz in the BW (Hz) box for the filter settings (this is the trade-off between PLL stability and noise rejection mentioned above) with a 4th order filter. Furthermore, the Q factor is of the order of $2 \cdot 10^3$ (as evaluated from the width of the resonance in the Bode plot in the frequency sweeper tab), and select the maximum peak to be 1 dB. After clicking on Advise, the Advisor will propose a 20 Hz/deg for the proportional and 2.58 ms for the integral gain respectively. Apply these parameters to PLL1 using the button in the Advisor: you are now ready to enable the PLL, which should now easily lock on the resonance.

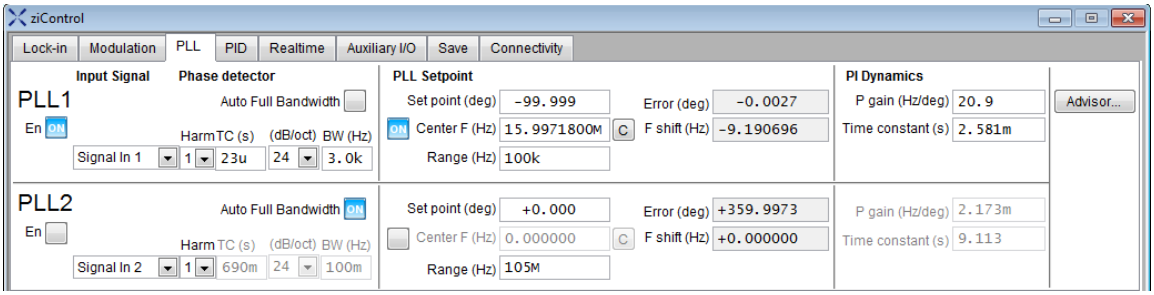


Figure 3.44. PLL tab with the parameters for the resonator

Users with the HF2LI-MF multi-frequency option could try to lock the second PLL on the third resonance, 3 dB weaker than the main one. The MF is needed to output the drive voltage from Output 1 instead of the default Output 2. (Users without MF could still make use of the second PLL, but they need to sum the excitation voltage from Output 2 to that of Output 1 by using the Signal Output 1 Add port; PLL2 can lock on the signal on Input 1, so there is no need to split the input signal from the resonator. A word of caution: there will also be an additional phase shift when using the Add port that should be kept into account when setting the PLL2 phase setpoint.)

The third resonance is much narrower than the first one and needs to be located with better precision: zoom in around it and find the frequency position of its peak. This specific resonator presents the resonance at 16.126 MHz, with a phase shift of -96°. Copy these values into PLL2 and apply the same parameters for the phase detector and the PID as for PLL1 (for instance by clicking on To PLL2 in the Advisor). Go to the Lock-in MF tab and enable Output 1 of demodulator 8 (the last one in the list), which is controlled by PLL2. Use an excitation voltage of 200 mV to drive the weaker resonance: Output 1 will contain a linear combination of the two excitation voltages with two different amplitudes. You can now enable both PLLs and they will lock on the two different resonances.

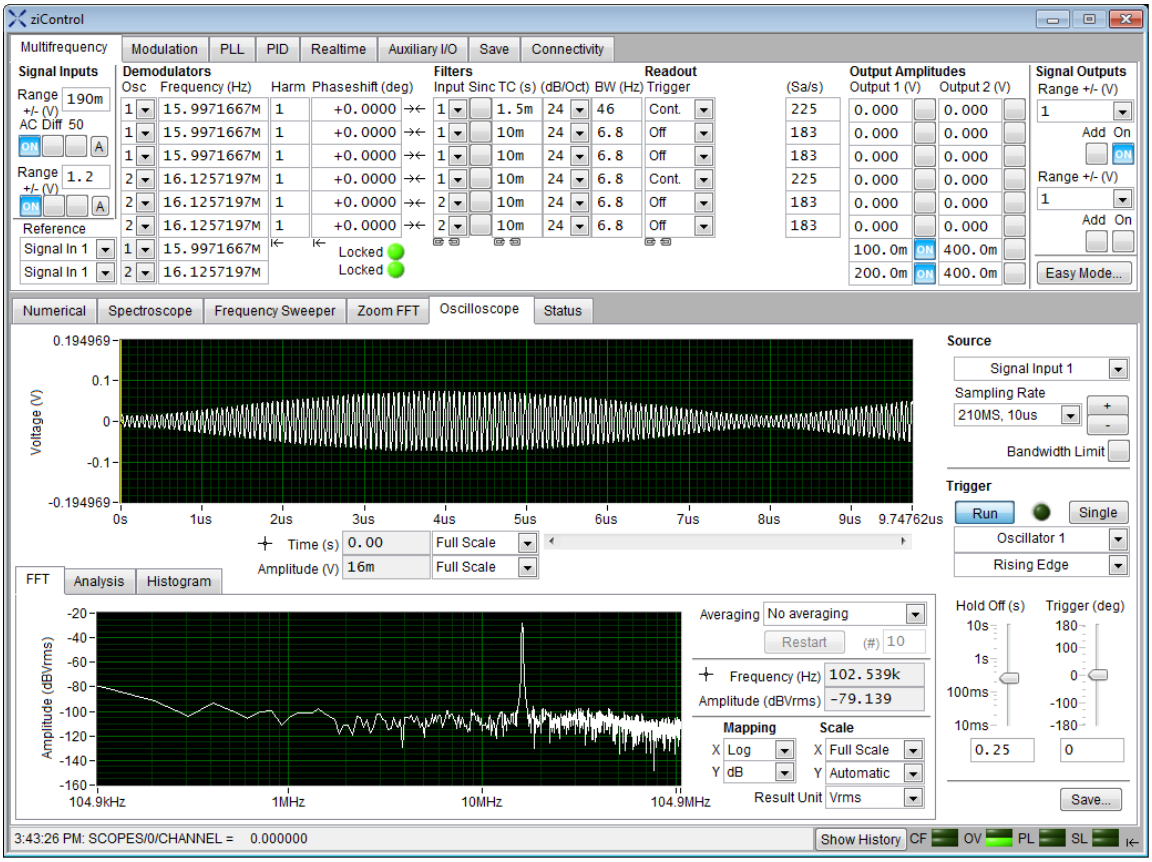


Figure 3.45. View of the Lock-in MF tab and the output signal in the Oscillator tab

Different PLL parameters could be used for the two resonances, provided the detection bandwidth of each PLL is small enough to suppress the signal from the nearby resonance. As an exercise, the user could use PLL1 to lock on the fourth resonance instead of on the first.

3.10. Tutorial PID Controller with Auto Tune

Note

This tutorial only applies to HF2 Instrument users that have purchased the HF2LI-PID option.

In this tutorial you will learn how to use a PID controller that calculates the required signal output amplitude in order to achieve a desired demodulator X value. You will then auto tune the controller's P, I and D gains to optimize the controller's performance.

First connect the signal output 1 connector to the signal input 1 connector with a BNC cable to create a simple feedback loop. Power cycle the HF2 Instrument in order to attain the instrument's default configuration and enable signal output 1 in the Lock-in tab of ziControl. Furthermore, select a first order (6 db/Oct) demodulator filter with a bandwidth of 1.00 Hz and a sampling rate of 3.6 kSamples/s for demodulator 1. This filter order and bandwidth create a slowly responding demodulator filter which will act as the system we aim to control. See Figure 3.46 for an over of the configuration in the Lock-in tab.

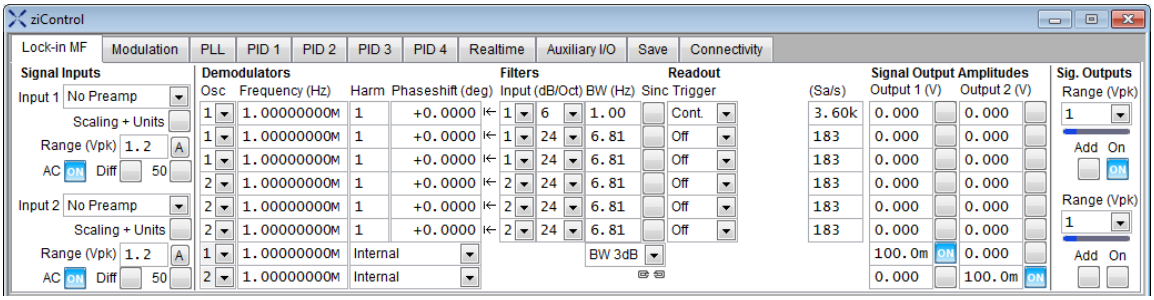


Figure 3.46. Lock-in tab configuration

In the PID 1 tab choose Demodulator: X on channel 1 as the controller's input, a Fixed setpoint of 100 mV_{RMS} and Signal 1: Amplitude on channel 7 as the controller's output. This corresponds to the default PID configuration which is shown in Figure 3.47 .

Switch to the Spectroscope tab in the lower half of ziControl and ensure that the spectroscope's display is set to plot the X value of demodulator 1. Now enable the PID controller, in the spectroscope you should see the demodulator's output approach 100 mV_{RMS} as in Figure 3.47 .

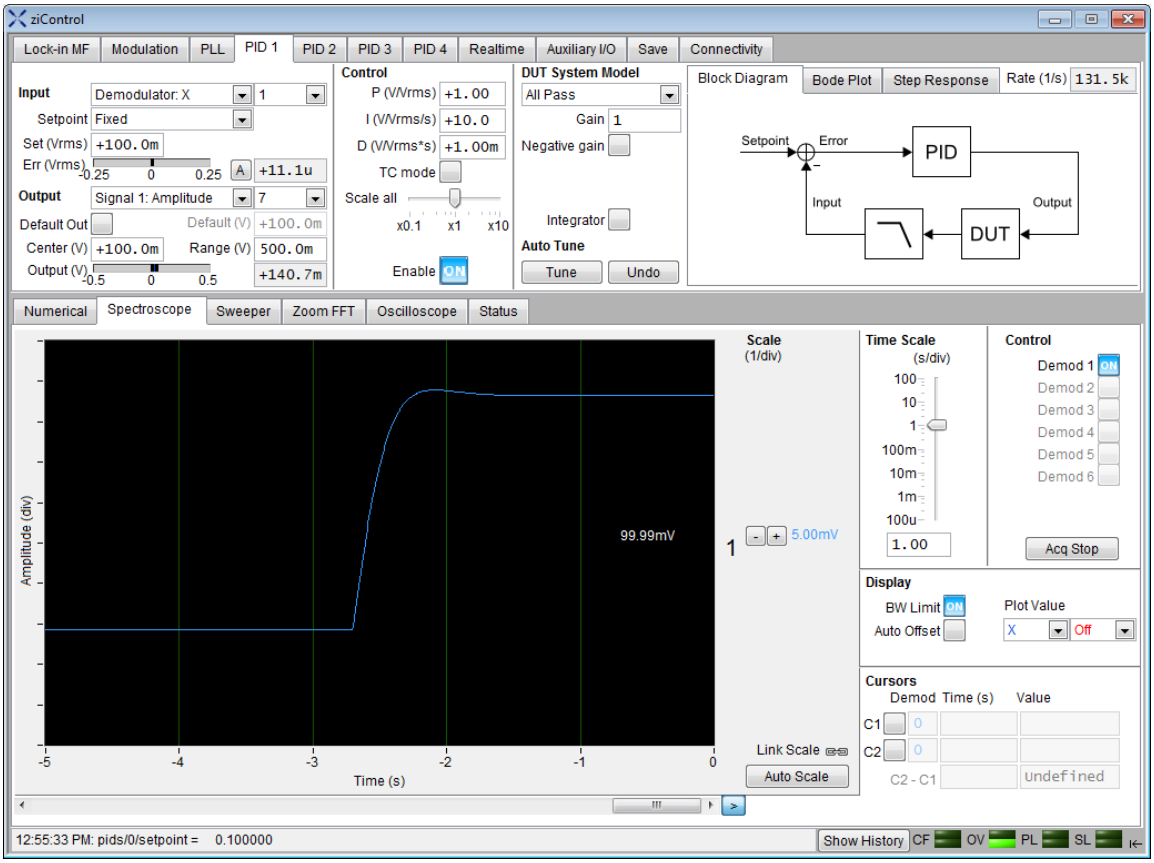


Figure 3.47. Initial PID configuration and spectroscope output with a fixed setpoint

In order to repeatedly view the controller's behavior with the current settings, change the setpoint to Toggle and set the two setpoints to the values 50 mV_{RMS} and 100 mV_{RMS} in the PID 1 tab. This configures the PID controller to toggle its setpoints between these two values at a rate of 1 Hz. After pressing Auto Scale in the Spectroscope tab you should see output similar to that in Figure 3.48

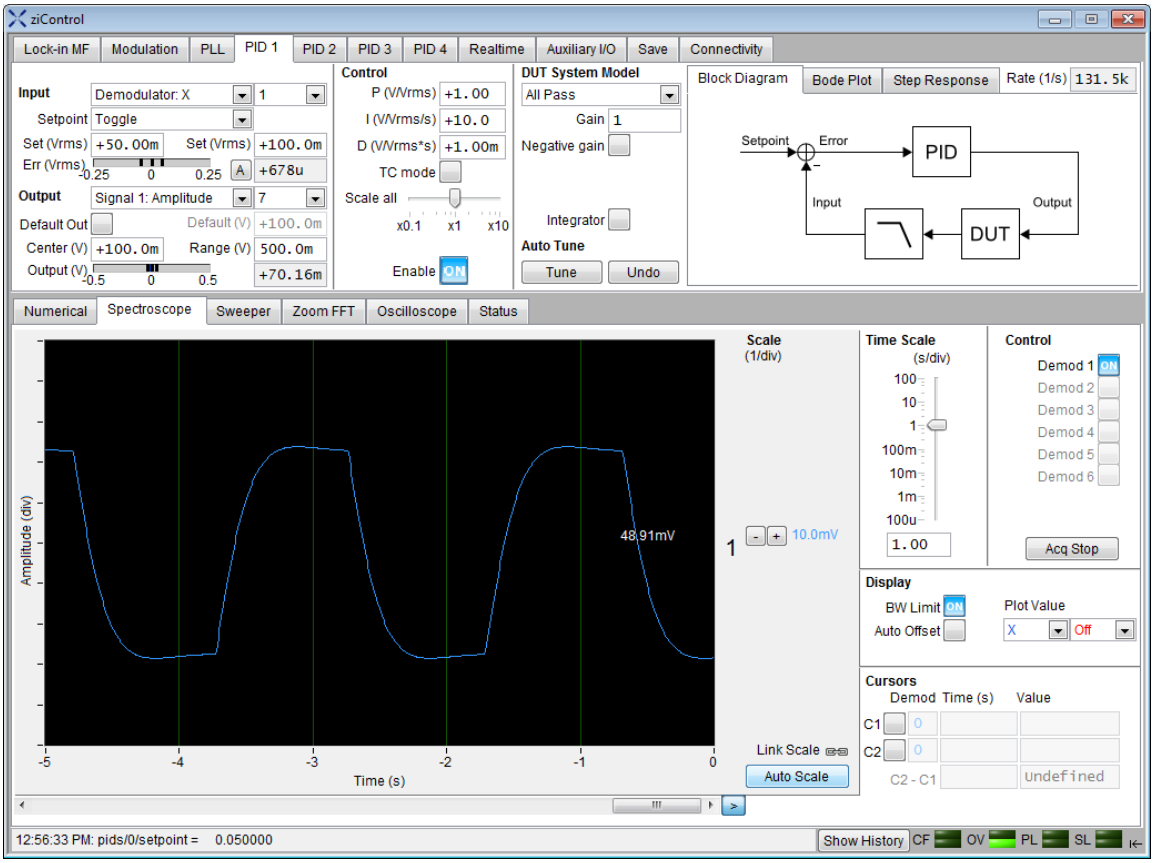


Figure 3.48. PID configuration and spectroscope output with two setpoints toggling between 50 mV_{RMS} and 100 mV_{RMS} using suboptimal PID control parameters

Now we'll optimize the PID controller's parameters using the Auto tune feature. Since we're using a simple feedback cable between signal output 1 and signal input 1, ensure that the System (DUT) Model used is All Pass and that the Gain is set to 707.1 in PID 1 tab, this models the cable (the DUT) appropriately as an all pass filter. The gain 707.1 corresponds to a gain of $1/\sqrt{2}$, which is needed due to fact that the PID input, demodulator X, is in RMS units (we assume the cable itself has a gain of 1). Now press the Tune button, in the Spectroscope tab you will see that the PID controller's response to the setpoint toggle is now much faster, see Figure 3.49 . The Auto tune feature has calculated optimal parameters for the controller using the Ziegler-Nichols method taking demodulator 1's filter values into account and configured the PID to use them.

In order to compare with the old controller parameters, you can press the Undo button in the Auto Tune section of the PID 1 tab. You can compare the response of the actual close loop in the spectroscope tab and the simulated step response calculated in the Step Response plot of the PID 1 tab.

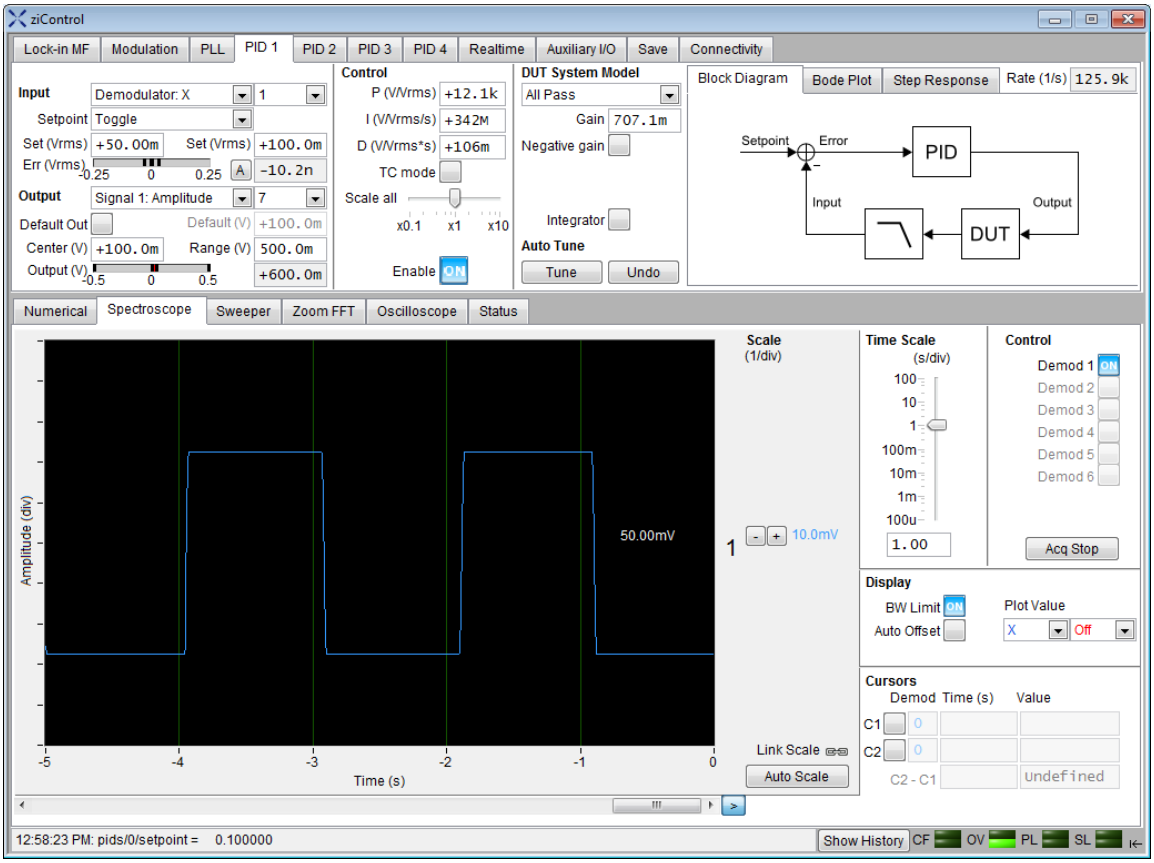


Figure 3.49. PID configuration and spectroscope output with setpoints toggling between 50 mV_{RMS} and 100 mV_{RMS} using optimal PID control parameters calculated with Auto tune

This concludes your PID controller with Auto tune tutorial. Thank you for measuring with Zurich Instruments.

3.11. Tutorial PID Controller Tuning Tools

Note

This tutorial only applies to HF2 Instrument users that have purchased the HF2LI-PID option.

The aim of this tutorial is to explain how to manually tune a PID controller's P, I and D gains by applying the Ziegler-Nichols method and to explain the PID tab's tuning tools. We will proceed by determining the PID parameters using the Ziegler-Nichols method to a simple control system by:

- manually determining the controller's gains by observing the output controller's response in the Spectroscope tab,
- more accurately and quickly determining the gains by using the Open Loop Bode plot in the PID tab,
- automatically determining the P, I and D gains using the Auto tune feature, which is partially based on the Ziegler-Nichols method.

3.11.1. The Ziegler-Nichols Closed-Loop Method

In the 1940s Ziegler and Nichols published a method for heuristically determining PID controller parameters that provide fast controller response whilst ensuring acceptable stability of the system. The basic idea is to experimentally extract information about the dynamics of the system that we wish to control. We do this by considering a proportional gain controller and searching for the P gain that brings this simpler control system to the limit of its stability. In general, if P is too small the system will be slow to respond to a change in the setpoint and if P is too large a change in the setpoint will cause the system to oscillate with ever increasing amplitude. We search for the value of P that creates oscillations in the system that neither decrease or increase upon a change in the set-point. We then use this P gain value, called the ultimate gain P_U, and the period of the induced oscillations T_U in order to calculate optimal P, I and D gains according to Table 3.25 . These values were determined by simulating and performing many experiments of common system processes found in industry.

Table 3.25. The values of P, I and D in terms of the critical gain P_U and corresponding oscillation period T_U according to the Ziegler-Nichols Method

P	I	D
$0.6*P_U$	$2*P/T_U$	$0.125*P*T_U$

After setting up the initial configuration according to Section 3.11.2 we'll apply the following steps of the Ziegler-Nichols method in Section 3.11.3 and Section 3.11.4 :

1. Set both the D and I gains of the controller to 0.
2. Increase the P gain, starting at 0, until we observe oscillations of constant amplitude in the value we desire to control. This value of P is called the critical gain P_U.
3. Measure the period T_U of the oscillations.
4. Calculate the controller's P, I and D gains according to Table 3.25 using P_U and T_U.

3.11.2. Lock-in Setup and Configuration

First connect the signal output 1 connector to the signal input 1 connector with a BNC cable to create a simple feedback loop. Power cycle the HF2 Instrument in order to attain the instrument's default configuration and enable signal output 1 in the Lock-in tab of ziControl. Furthermore, select a fourth order (24 db/Oct) demodulator filter with a bandwidth of 6.81 Hz and a sampling rate of 14.4 kSamples/s for demodulator 1. These are the default demodulator settings which together with a feedback cable act as the system we aim to control. See Figure 3.46 for an overview of the configuration in the Lock-in tab.

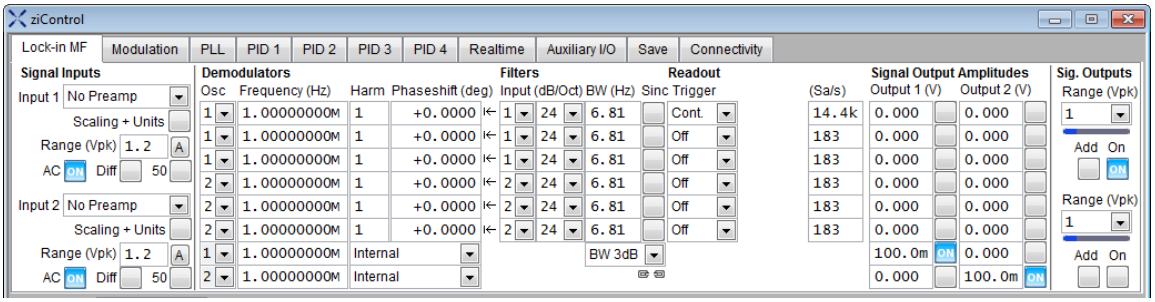


Figure 3.50. Lock-in tab configuration.

3.11.3. Determining the PID Parameters Manually

In this section we determine the controller's P, I and D gains by applying the Ziegler-Nichols method manually, without using the PID tab's Auto tune feature.

First, we appropriately configure the controller for Ziegler-Nichols. In order to observe the system's response to a change in the setpoint of the PID we change the Setpoint to Toggle and set the two setpoints to 90 V_{RMS} and 100 V_{RMS}. This will change the PID setpoint between these two values at a rate of 1 Hz. For an overview of the PID tab's configuration see Figure 3.51 .

Note

We choose a relatively small change in the two setpoints in order to help avoid saturating the controller's outputs. A large value of P drives the outputs hard, causing them to either reach the user-imposed PID limits (specified via center and range) or the hardware limits of the HF2. If we're reaching the output limits, it's more difficult to tell whether the oscillations in the system really have constant amplitude or whether the outputs are simply saturated.

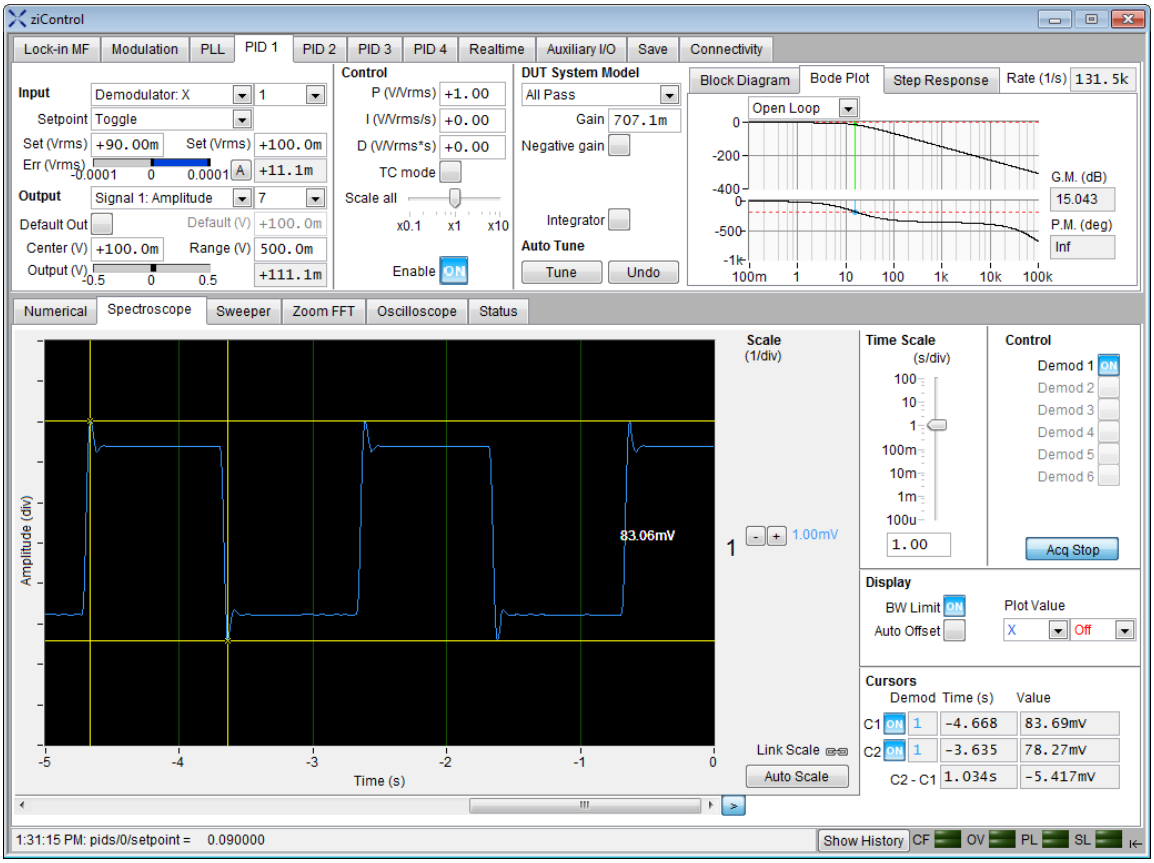


Figure 3.51. Spectroscope output and initial PID configuration with two setpoints toggling between 90 mV_{RMS} and 100 mV_{RMS}. With P=1, I=0, D=0 the controller performs poorly achieving neither upper nor lower setpoint. Since there is only a small overshoot upon a setpoint change, we have to increase P in order to incite oscillatory behavior

We now apply the steps detailed in Section 3.11.1 to our simple control system:

1. Set both the D and I gains of the controller to 0.
2. Increase the P gain, starting at 0, until oscillations of constant amplitude are observed in the demodulator X value. We see in Figure 3.51 that for P=1 oscillations are not yet observed, but rather a small overshoot from the controller's steady state. When P=5.00 we clearly see that the system exhibits oscillatory behavior and for P=5.70 we see that the system becomes instable. Now, we finely tune P to obtain oscillations with constant amplitude for and see that P=5.61 is a suitable value for the critical gain P_U, see Figure 3.52 .
3. Now we measure the period T_U of the oscillations in the spectroscope tab. First zoom in by adjusting the Time Scale so that a single oscillation can be clearly viewed in the spectroscope. Then enable the cursors C1 and C2 and move them so that the vertical bars correspond to one period of the oscillations. The difference between the two cursors C2-C1 corresponds to T_U, see Figure 3.53 .
4. Finally, we calculate the controller's P, I and D gains according to Table 3.25 with P_U=5.61 V_{RMS} and T_U=64 ms and find that we should set the controller's gains as P=3.37, I=105 and D=0.0269.

We see that with these gain values the controller does overshoot but quickly settles to the setpoint. See Figure 3.54 to see the result of the tuned controller.

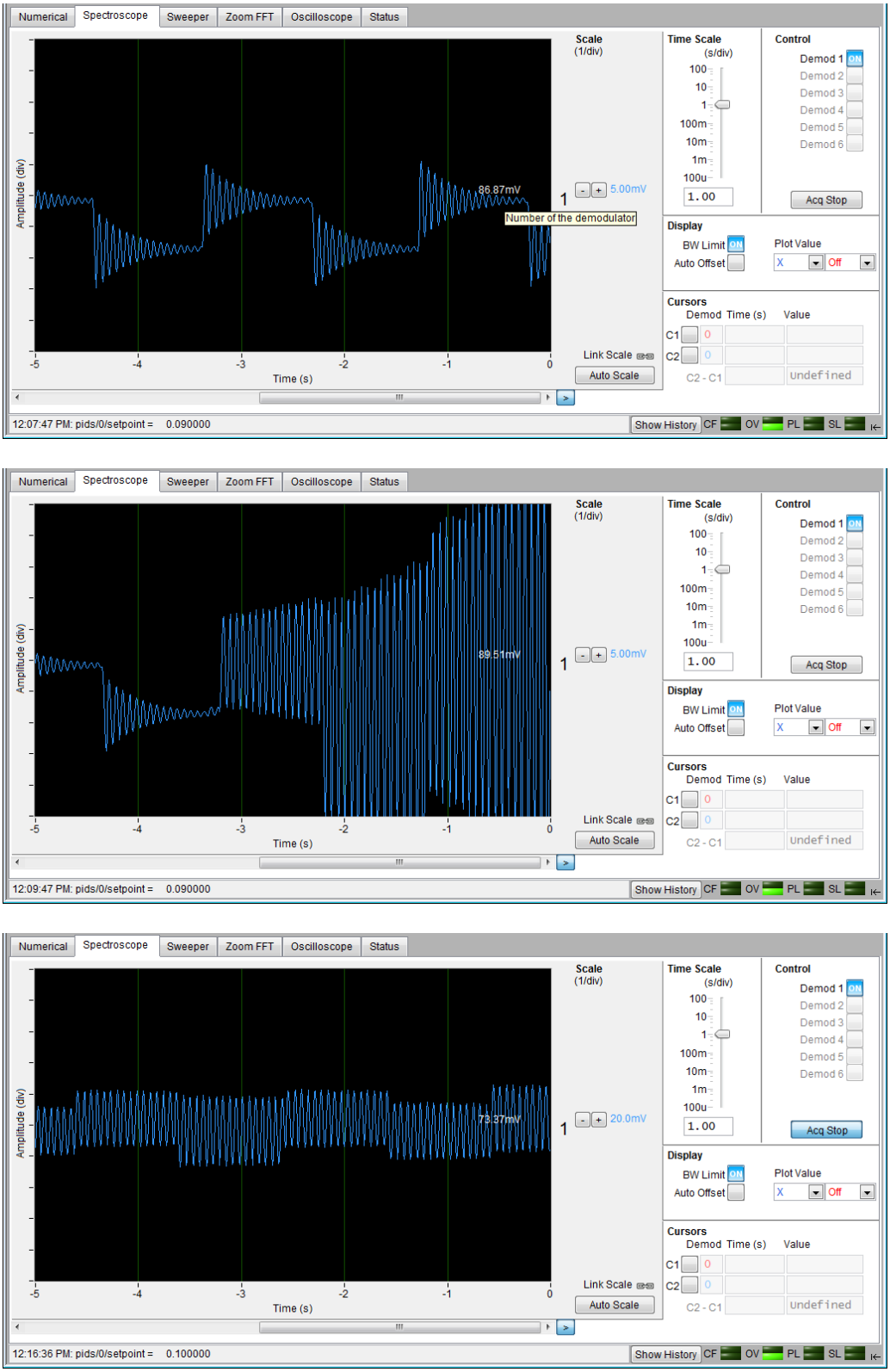


Figure 3.52. Three spectroscope snapshots with P=5.00, 5.70, 5.61 (top,middle,bottom) showing demodulator output when the PID is configured to toggle between two setpoints, 90 mV_{RMS} and 100 mV_{RMS}, both I and D are kept equal to 0. The top snapshot demonstrates stable system behaviour with P=5.00, in the middle shot P has been increased to 5.70 and demonstrates instable behaviour. The bottom snapshot shows stable oscillations which neither increase nor decrease in amplitude over time, here P=5.61 which corresponds approximately to the critical gain P_U

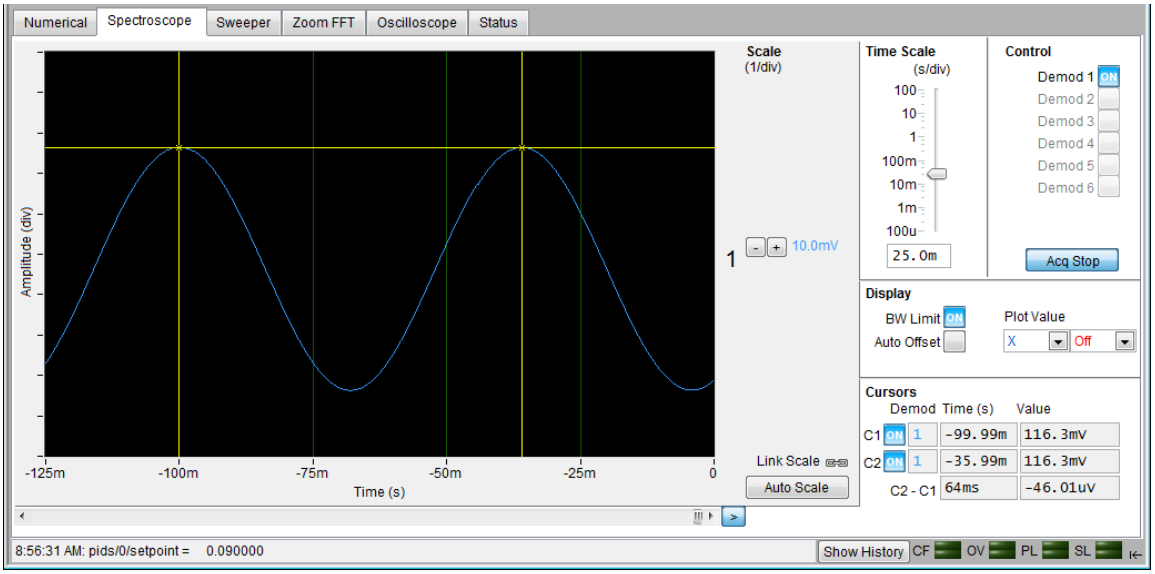


Figure 3.53. After we're satisfied with the value we've found for the critical gain P_U, we zoom in on a spectroscope snapshot in order to measure the period of the oscillations, T_U. Using the cursor tools we see that for P_U=5.61 the corresponding period is T_U=64.0 ms

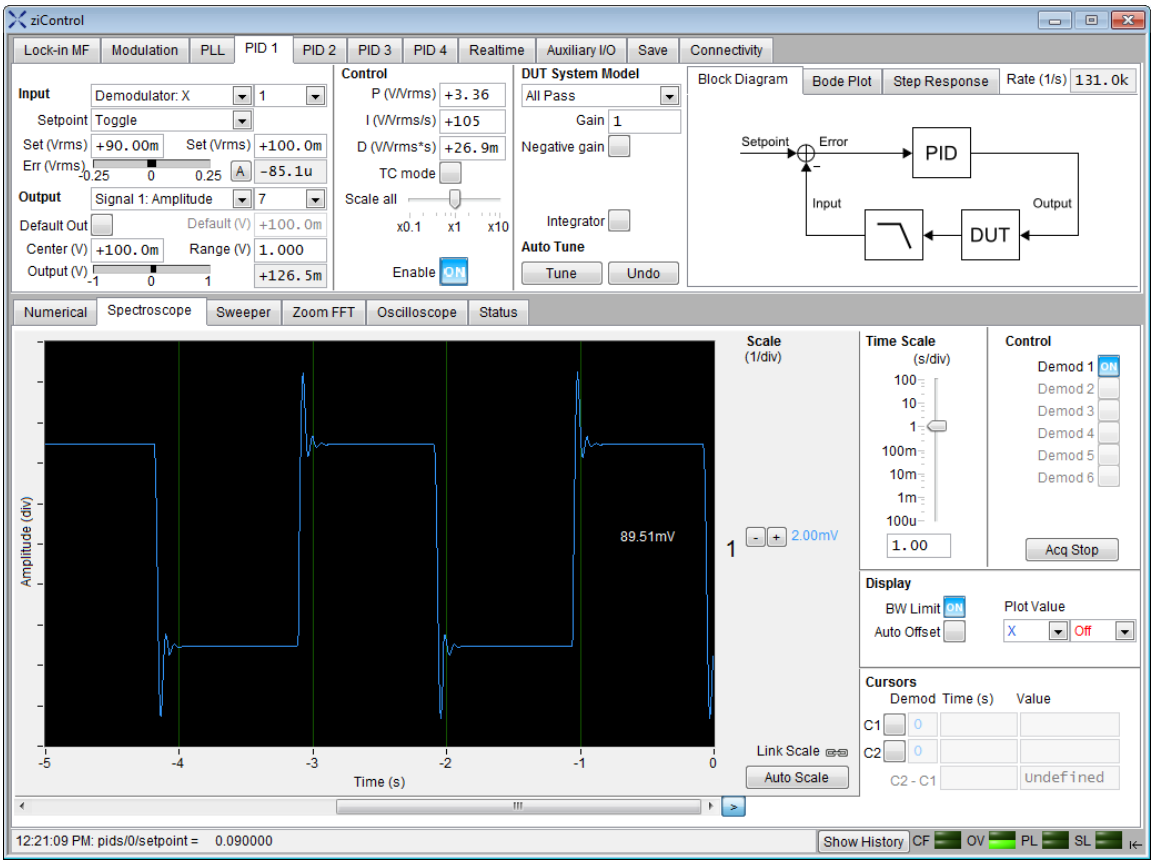


Figure 3.54. The tuned PID configuration and spectroscope snapshot

3.11.4. Determining the PID Parameters Manually using the Open Loop Bode Plot

In this section we apply the Ziegler-Nichols method using the tuning tools available in the PID tab. To use these tools we first have to provide a system model that describes the device under test. We specify the model in the DUT System model section of the PID tab. Since our device under test is a simple feedback cable, a suitable model for our system is the All Pass model and we set the gain to 0.7071. This gain corresponds to $1/\sqrt{2}$, appropriate since the PID input, demodulator X, is in RMS units (we assume the cable itself has a gain of 1). We don't need to select Negative Gain or add an Integrator to the model. See Figure 3.55 for a screenshot of the model's parameters in the PID tab.

In the Bode Plot sub-tab of the PID tab the user can plot the transfer functions of various components of the system or the entire system itself. In particular we can plot the transfer functions of:

- PID
- System model for the Device Under Test (DUT)
- Demodulator filter (updated with the current settings)
- Combined system model and demodulator filter
- Open loop of the entire system: the system model, demodulator filter and PID controller
- Closed loop of the entire system: the system model, demodulator filter and PID controller; this describes the full feedback system

Note

With the exception of the PID and demodulator filter Bode plots, the reliability of the bode plots depends on the accuracy of the provided model for the DUT.

In the following, we'll make use of the Open Loop Bode Plot to determine the critical gain. The Open Loop plot can be used to tell whether the feedback system with the current P, I and D gains is stable or not. The top plot is the Bode magnitude plot and the bottom the Bode phase plot. By the definition of feedback systems, the closed loop system is instable when the open loop gain is greater than 1 at the frequency where the open loop phase is -180°.

The phase margin and gain margin both give an indication of the proximity to instability. The phase margin is defined as the amount of phase lag required before the closed loop system becomes instable. Similarly, the gain margin is defined as the smallest amount that the open loop gain can be increased before the closed loop system becomes instable. The blue bar in the Open Loop Bode magnitude plot indicates the frequency where the gain margin is 0 dB (corresponding to a gain of 1 in linear units) and the blue bar in the Open Loop Bode phase plot indicates the frequency where the phase margin is zero. The actual phase margin (P.M.) and gain margin (G.M.) of the feedback system are displayed at the right-hand side of the Open Loop bode plot and are highlighted in green in the Open Loop Bode plots. The distance along the green line between the dashed red line and the black curve of the bode plot gives the gain margin, respectively, phase margin.

By definition, a gain margin of 0 dB corresponds to the point after which the system becomes instable, and if we set I and D to zero, then the value of P that produces an open loop gain margin of 0 dB is exactly the critical gain P_U. This is visible in the open loop Bode plot as when the phase margin and gain margins can be found at the same frequency and the blue and green lines lie on top of one another. We now proceed to find this point experimentally by increasing the controller's P gain.

We start with the same configuration that we used in Section 3.11.3 and perform the steps of the Ziegler-Nichols method, this time choosing P_U based on the gain margin as displayed in the Open Loop plot:

1. Set both the D and I gains of the controller to 0.
2. Increase the P gain from 0 until the gain margin in the Open Loop Bode plot changes sign from positive to negative. Ideally we want to choose the value of P that creates a gain margin of 0 dB. See Figure 3.55 for Open Loop bode plots that correspond to stable, instable and critically stable closed loop systems. We obtain a gain margin of 0 when P=5.6059.
3. Measure the period T_U of the oscillations in the spectroscope tab as in the previous section, shown in Figure 3.53 . We again find that T_U=64 ms. Note, if the supplied DUT model is incorrect the simulated behavior and actual system behavior will not coincide and you will not be able to measure T_U.
4. Finally, we calculate the controller's P, I and D gains according to Table 3.25 and find that we should set the controller's gains as P=3.364, I=105.1 D=0.02690.

Figure 3.55. With P=2,20,5.6059 we obtain gain margins of 9.022,-10.978 and 0.000 dB respectively. These values correspond to stable, instable and critically stable closed loop systems.

Note

The axes limits in the Bode plots are calculated automatically, but it's possible to change them by double clicking on the current axes limit value, typing a new value and pressing [ENTER].

Note

We can also use the Closed Loop Bode plot to see whether the PID controller is stable or not: To the right-hand side of the plot there is a Stability Issue lamp which turns red if the current PID gains result in an instable feedback system, see Figure 3.56 .

Figure 3.56. The PID tab's Closed Loop Bode plot includes a lamp indicating whether the simulated feedback system is stable or not. As we've seen for P=5.605 the system is stable, whereas for P=5.660 it's instable

3.11.5. Determining the PID Parameters using Auto tune

If the specified DUT system model is correct, we can use the Auto tune feature to calculate the PID gains automatically. After providing the model as in Section 3.11.4 , Auto tune determines the PID gains to be P=3.3908, I=106.13 and D=0.027082. The tuned PID is shown in Figure 3.57 . Since Auto tune is partially based on the Ziegler-Nichols method the P, I and D gains are very close to those determined manually in Section 3.11.3 .

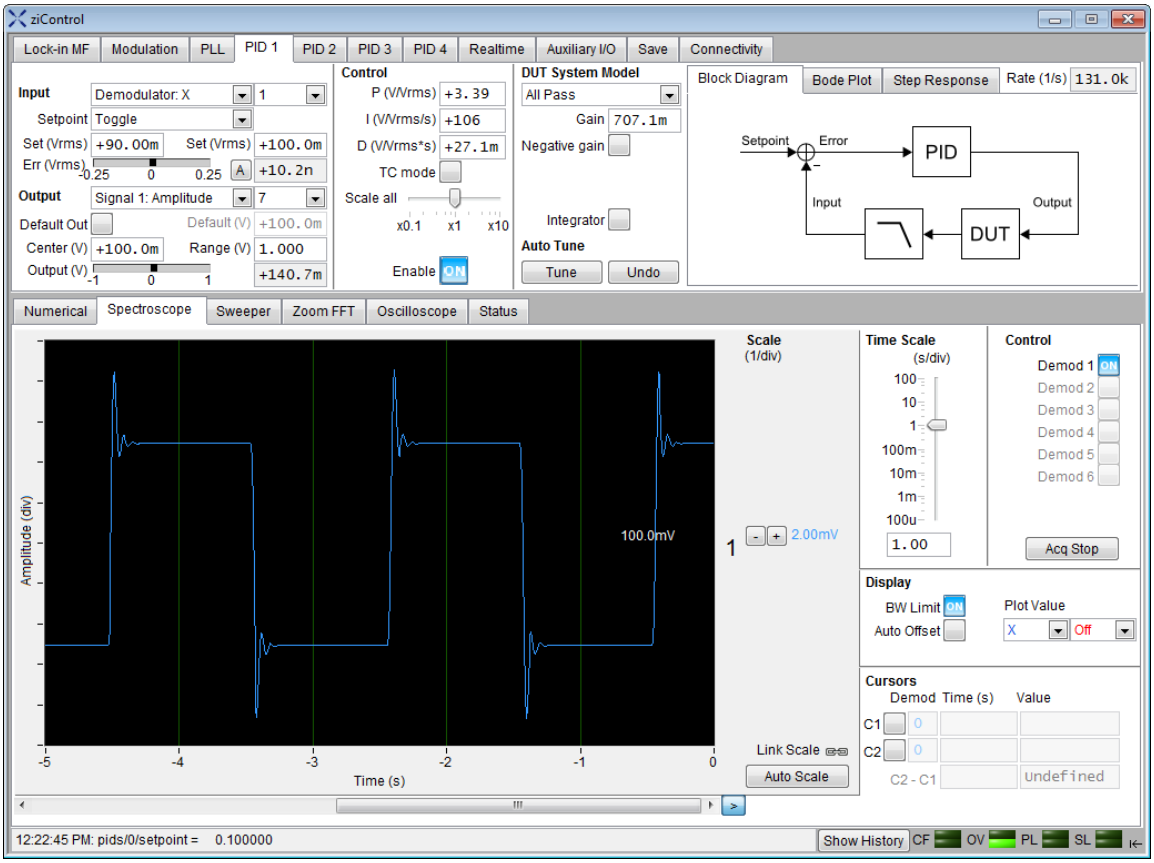


Figure 3.57. The tuned PID and output in the Spectroscope after applying Auto tune

Note

We can view the simulated step response of our system in the Closed Loop Step Response Plot. Under the assumption of an accurate model, this allows us to view the system's response without enabling the PID, an example is shown in Figure 3.58 .

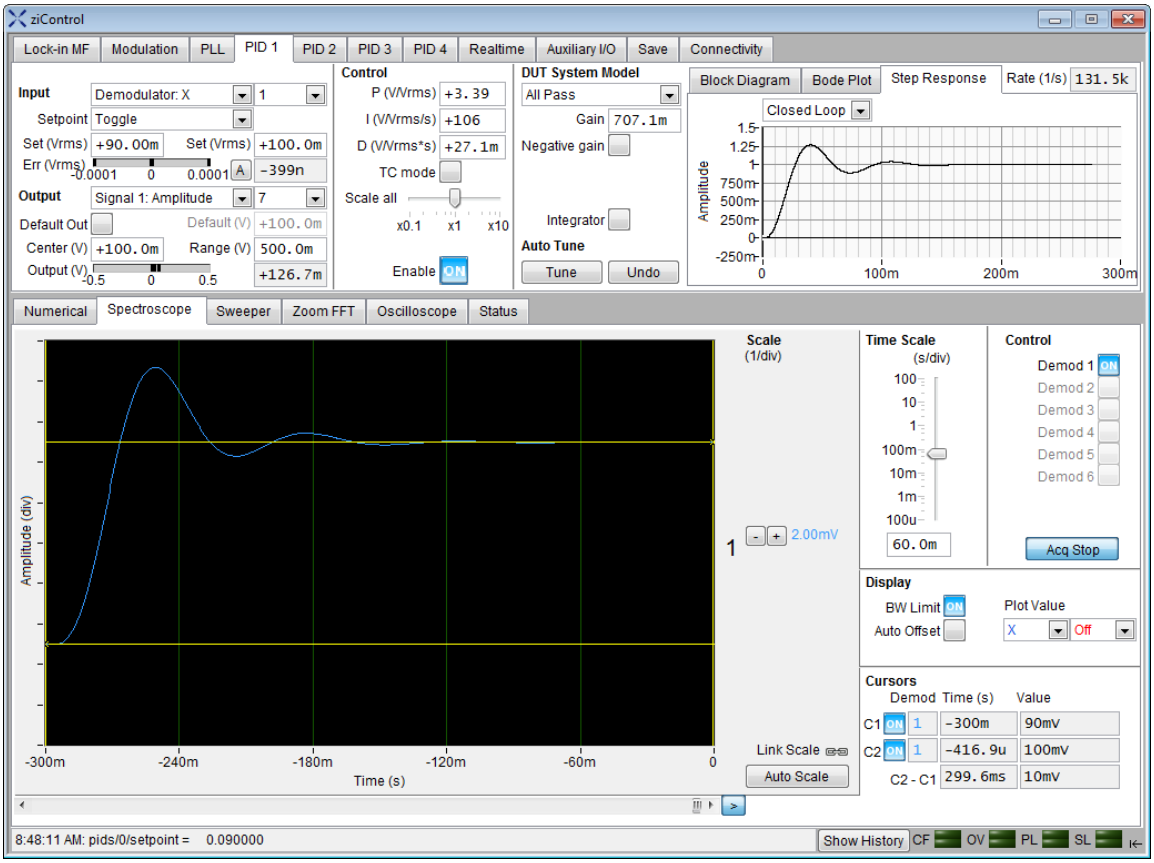


Figure 3.58. The PID tuned via Auto tune. The simulated closed loop step response coincides well with the true response shown in the Spectroscope.

Chapter 4. Functional Description HF2LI

This section contains the detailed description of all panels of the graphical user interface (GUI) ziControl for the HF2LI and HF2PLL products. This GUI is a LabVIEW based program that is delivered standard with all instruments.

On top of standard functionality like acquiring and saving data points this GUI provides a variety of measurement tools for the time and frequency domain. All of these features (and a few more) are also accessible by means of the programming interfaces described in Chapter Programming .

4.1. Graphical User Interface Overview

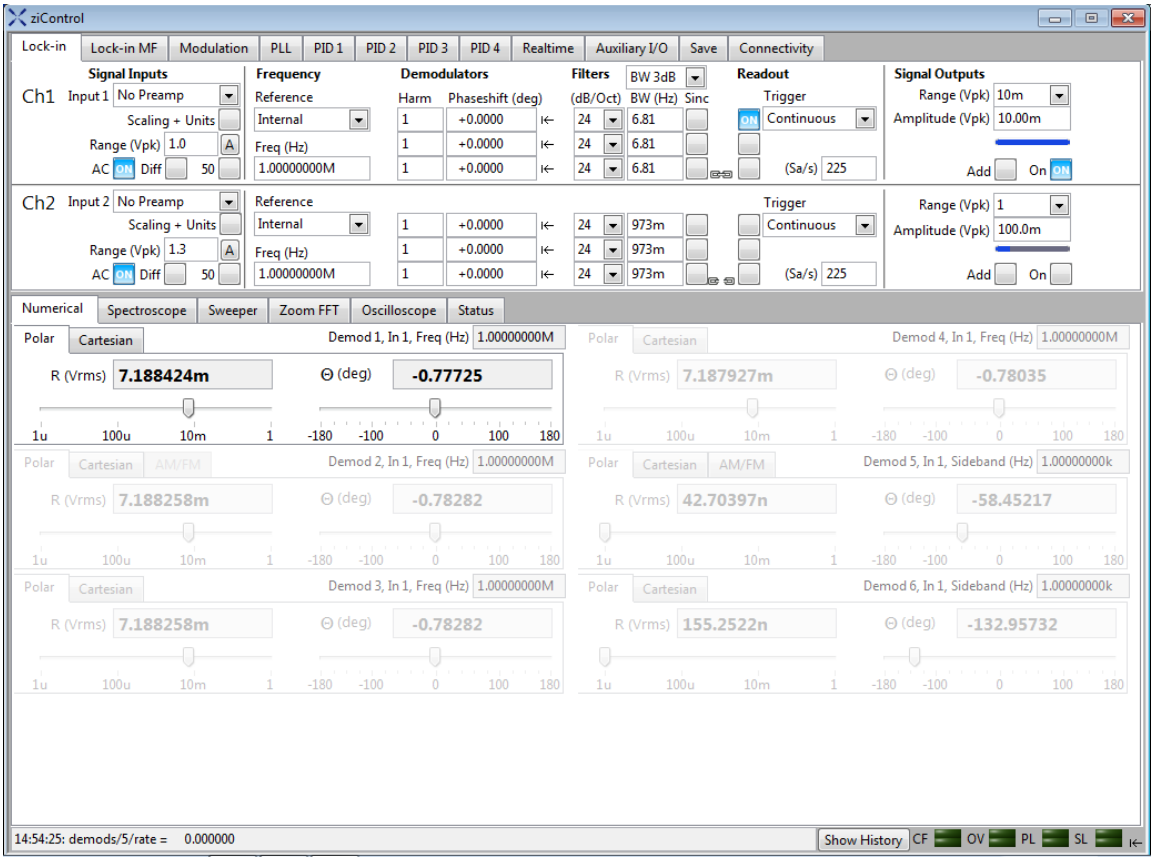


Figure 4.1. ziControl Overview

The GUI is divided into 2 sections as depicted in Figure 4.1 , each is subdivided in a tab structure. The **settings section** on top is dedicated to display and control the main settings of the instrument, whereas the **tools section** provides additional measurement functionality. A status line on the lower end shows the command history which ziControl exchanged with the ziServer plus a couple of status flags on the right hand side.

The following tabs are available in the settings section:

- Lock-in
- Lock-in MF (available as upgrade option)
- Modulation (available as upgrade option)
- PLL (available as upgrade option)
- PID (available as upgrade option)
- Real-time (available as upgrade option)
- Auxiliary I/O
- Save
- Connectivity
- Active probes (displayed only when optional additional hardware is attached)

These are the tabs in the tools section:

- Numerical
- Spectroscope
- Frequency response sweeper
- ZoomFFT
- Oscilloscope
- Status

4.2. Settings Tabs

4.2.1. Lock-in Tab

Note

This tab is the lock-in control panel for all base instruments. This tab is not available for instruments with installed HF2LI-MF option.

Features:

- Control for 2 separate lock-in units with 3 demodulators each
- Auto ranging, scaling and definition of units for each input channel
- Control for 2 signal generators
- Range setting for signal inputs and signal outputs
- Flexible choice of reference source and trigger options
- Automated scale and unit adjustment for external amplifier

Description:

The Lock-in tab (Figure 4.2) is divided into 2 identical sections, separated by the horizontal line in the middle. The upper section is related to signal input 1 and output 1, and the lower section to signal input 2 and output 2.

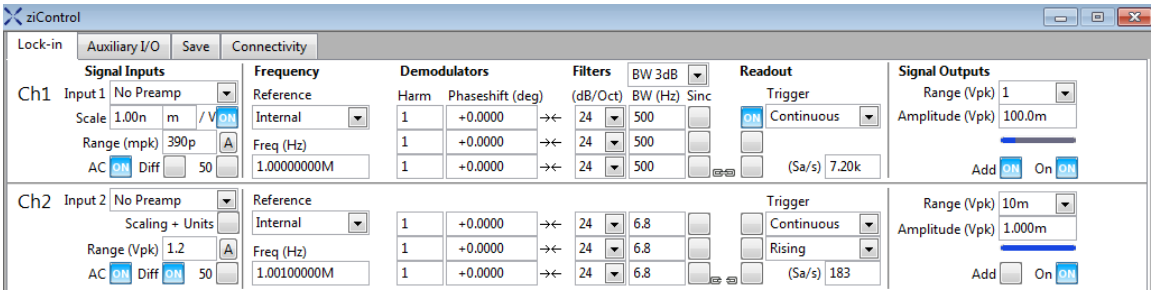


Figure 4.2. Lock-in Tab

Table 4.1. Lock-in Tab

Control/Tool	Options/Range	Description
Input selection	No Preamp	defines the active probe that is connected to the input of the HF2 instrument - in case an HF2 Series pre-amplifier is connected then this selection will make the measurement account for external pre-amplification and current conversion - please make sure to define the correct pre-amplifier model, channel, and ZCtrl connectivity port (on the back panel of the instrument)
	HF2TA 0, Ch1	
	HF2TA 0, Ch2	
	HF2CA 0	
	HF2TA 1, Ch1	
	HF2TA 1, Ch2	
	HF2CA 1	
Scale + Unit enable	OFF / ON	enables arbitrary input scaling to define a scaling factor for all measurements regarding this input channel and also change the unit which is displayed throughout the tools and settings

Control/Tool	Options/Range	Description
Scale factor	numeric scaling factor	positive or negative real number to scale all measurement values attributed to that particular input; only appears when Scaling + Units is enabled
Unit acronym	three letter acronym	three letters can be set to define a unit, e.g. PSI; only appears when Scaling + Units is enabled
Range (unit)	1 mV to 2.1 V (or equivalent after unit + scale is applied)	sets the range of input 1/2 in fine increments within the available range - the value is the absolute range of the signal including a potential DC offset - note: the value inserted by the user may be approximated to the nearest value supported by the HF2; Scaling + Units is fully accounted for in this setting
A = Auto Range button	[press once]	press this button to automatically set the input range to two times the maximum amplitude of the input signal over a measured time of 100 ms
AC switch	OFF: DC coupling	sets the input coupling for input 1/2 - an AC coupling inserts a high-pass filter at 1 kHz in the input path
	ON: AC coupling	
Differential switch	OFF: single-ended input	selects whether to connect to a single-ended signal (one cable) or to a differential signal (two cables)
	ON: differential input	
50 Ω switch	OFF: 1 MΩ	sets the matching impedance for input 1/2
	ON: 50 Ω	
Reference selection	Internal (internal reference)	internal reference mode: the lock-in reference is generated by internal oscillators auto reference mode: the lock-in reference is derived from the input signal external reference mode: the lock-in reference is generated externally and connected to the indicated connector other internal reference mode: the lock-in reference is the same as for the other lock-in unit (variation of the internal reference mode) note: frequency restrictions apply for some selections
	Signal In 1 (auto/ external reference)	
	Signal In 2 (external/ auto reference)	
	Signal Aux In 1 (external reference)	
	Signal Aux In 2 (external reference)	
	Signal DIO 0 (external reference)	
	Signal DIO 1 (external reference)	
	From Ch1 / Ch2 (other internal reference)	
Lock flag	OFF / ON	this flag indicates when the oscillator is locked to the selected reference frequency - note: this flag is only meaningful when external and auto reference modes are selected

Control/Tool	Options/Range	Description
Freq (Hz)	0 to 100 MHz	frequency for internal reference 1/2: this frequency can also be output with variable amplitude
Harm (1..6)	1 to 1023	selects the harmonic to be measured - enter 1 to measure the fundamental frequency and 2 for the 2nd harmonic, and so on
Phaseshift (1..6)	-180° to 180°	defines the phase shift of the reference frequency at the input of the demodulator (works for internal and external reference modes) - this setting does not shift the signal output
Auto Zero button ⇤	[press once]	this button shifts the phase of the reference at the input of the demodulator in order to achieve 0 phase at the demodulator output - this action maximizes the X output, zeros the Y output, zeros the Θ output, and leaves the R output unchanged
Filter Roll Off selection	1st: 6 dB/oct	sets the filter roll off (filter order) for the related demodulator
	2nd: 12 dB/oct	
	3rd: 18 dB/oct	
	4th: 24 dB/oct	
	5th: 30 dB/oct	
	6th: 36 dB/oct	
	7th: 42 dB/oct	
	8th: 48 dB/oct	
Filter Property Unit selection	BW 3 dB: 80 µHz (filter order = 8) to 200 kHz (filter order = 1)	defines the filter properties displayed in either of these bandwidth or time constant units that differ by a scaling factor depending on the filter order - please refer to Table 10.1 for exact relation between TC, BW, and filter order - note: the value inserted by the user may be approximated to the nearest value supported by the HF2
	BW NEP: 90 µHz to 319 kHz	
	TC eff: 783 ns to 1900 s	
	TC per order: 783 ns to 580 s	
Sinc Filter switch (1..6)	OFF: Sinc filter disabled for corresponding demodulator	the Sinc filter is an additional filtering stage that permits to remove the omega and 2 times omega components - by using a large filter roll-off value, the user can effectively reduce the signal component at the second and third harmonics - for applications where large roll-off is not possible, the Sinc filter achieves the same effect - note: there are limitations regarding the maximum frequency and the frequency resolution allowed with Sinc filtering (see Table 4.2)
	ON: Sinc filter enabled for corresponding demodulator	
Block Lock button	[press once]	this button locks the settings between the different demodulators - when lock is closed, then changing one value changes all others - when lock is open, changing one leaves others unchanged

Control/Tool	Options/Range	Description
Demodulator switch (1..6)	ON / OFF	switches to turn on and off each of the demodulators
Trigger selection	Continuous	automatic internal trigger with the sample rate defined in the next control field
	DIO 0	samples are sent to the host computer depending on DIO 0 triggering
	DIO 1	samples are sent to the host computer depending on DIO 1 triggering
	DIO 0 or 1	samples are sent to the host computer depending on DIO 0 and 1 triggering
	Undefined	a complex trigger has been programmed to the HF2 Instrument by means of the programming interfaces - this mode cannot be selected by the user of ziControl, but is the result of a complex trigger definition - note: this is a read-only position
Trigger Mode selection	Rising Edge	1 data sample is sent to the host computer for a rising edge on the trigger input
	Falling Edge	1 data sample is sent to the host computer for a falling edge on the trigger input
	Both Edges	1 data sample is sent to the host computer for a rising edge and falling edge on the trigger input
	Gate High	data samples are sent to the host computer as long as the defined DIO 0/1 input is high - it is possible to determine the number of demodulated samples that are sent to the host computer after a trigger by setting the readout rate and modulating the pulse length on the DIO 0/1
	Gate Low	data samples are sent to the host computer as long as the defined DIO 0/1 input is low - it is possible to determine the number of demodulated samples that are sent to the host computer after a trigger by setting the readout rate and modulating the pulse length on the DIO 0/1
	Undefined	a complex trigger has been programmed to the HF2 Instrument by means of the programming interfaces - this mode cannot be selected by the user of ziControl, but is the result of a complex trigger definition - note: this is a read-only position
Sampling Rate	0.22 Hz to 460 kHz	sets the filter readout rate, equivalent to the demodulator samples (measured data points) that are sent to the host computer per second - it is also the rate of data received by ziControl and saved to the computer hard disk - this setting has no impact on the sample rate on the physical auxiliary outputs - note: the value inserted by the

Control/Tool	Options/Range	Description
		user may be approximated to the nearest value supported by the HF2 (see Table 9.4)
Range selector	±10 mV	sets the output range for output 1/2 - this select determines the maximum output peak to peak range - this setting will make sure that no peaks above the setting are generated at the output, independent from the amplitude settings (V) - the output signal is clipped if the amplitude is higher than the range - if the range is changed to a value smaller than V, then V is automatically reduced to the new range
	±100 mV	
	±1 V	
	±10 V	
Amplitude (V_{pk})	0 to 10 V	sets the output voltage - the minimum setting is 0 V, the next level above zero depends on the signal output 1/2 range - note: the value inserted by the user may be approximated to the nearest value supported by the HF2
Amplitude indicator	0 to 100%	graphical indicator on how much of the set output range is currently used
Add switch	OFF: Add disabled	the Add input allows to add an external analog signal to the internally generated signal - when disabled, the signal at the Add input is ignored
	ON: Add enabled	
Output switch	OFF: output disabled	sets the main switch for output 1/2 - when the output is enabled, then the HF2 signal output is active and the blue LED on the front panel lights-up
	ON: output enabled	

For technical reasons, the Sinc filter imposes restrictions on possible demodulation frequencies. This restriction is important in the 10 kHz range, whereas it is small in the 10 Hz range, where most applications require the Sinc filter operate.

Table 4.2. Sinc filtering range and setting resolution

Demodulation frequency	Frequency setting resolution	Comments
Above 10 kHz	not supported	the frequency resolution depends on the absolute value when Sinc filtering is enabled
10 kHz	220 Hz	
1 kHz	2.2 Hz	
100 Hz	0.022 Hz	
10 Hz	0.0002 Hz	
1 Hz	2 µHz	

4.2.2. Lock-in MF Tab

Note

Instruments with the HF2LI-MF option installed will show the Lock-in MF tab instead of the Lock-in tab. The HF2LI-MF option can be bought and installed at any time via remote servicing.

Compared to the basic version of the instrument the multi-frequency option offers the use of 6 independent numerical oscillators where individual output amplitudes can be defined for each of the two outputs.

Features:

- Control for 6 independent demodulation units for 2 input signals
- Auto ranging, scaling and definition of units for each input channel
- Control for 8 oscillators for internal reference demodulation and signal generation
- Flexible choice of reference source and trigger options
- Arbitrary input units and scaling for adjustment to external transducers

Description:

The HF2LI-MF option is delivered with a tab that permits to control all settings additional compared to the base lock-in amplifier. It is a crowded tab providing a very large flexibility with the settings. It is possible to achieve overall settings with various combinations.

Each line in the Demodulators section permits to control the inputs and the settings of one of the 6 demodulators. For instance it is possible to connect one of the 8 oscillators (Osc) to any of the Demodulators, permitting to a very flexible frequency allocation, e.g. any line in the tab can be configured to work with any oscillator. Line 7 and 8 are a bit different as the oscillator allocation 1 and 2 is given and cannot be changed.

On the signal generation side, it is easy to see that it is possible to generate Output 1 and Output 2 as a linear combination of up to 8 frequencies generated with said oscillators.

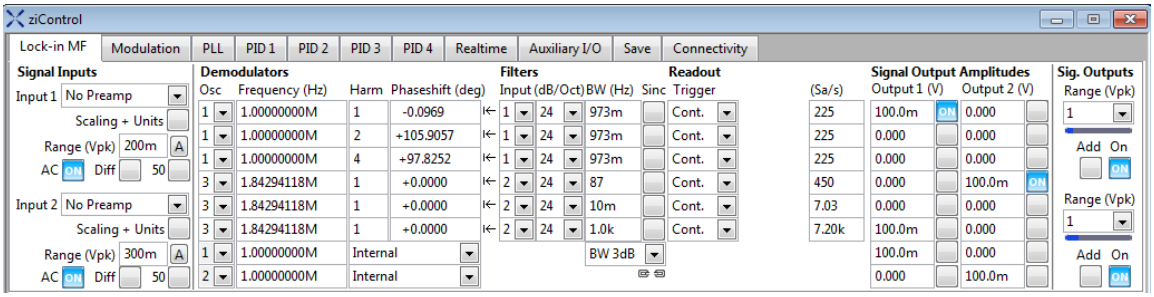


Figure 4.3. Lock-in MF Tab

Table 4.3. Lock-in MF Tab

Control/Tool	Options/Range	Description
Input selection	No Preamp	defines the active probe that is connected to the input of the HF2 instrument - in case an HF2 Series pre-amplifier is connected then this selection will make the measurement account for external pre-amplification and current conversion - please make sure to define the correct pre-amplifier model, channel, and ZCtrl connectivity port (on the back panel of the instrument)
	HF2TA 0, Ch1	
	HF2TA 0, Ch2	
	HF2CA 0	
	HF2TA 1, Ch1	
	HF2TA 1, Ch2	

Control/Tool	Options/Range	Description
	HF2CA 1	
Scaling + Units enable	OFF / ON	enables arbitrary input scaling to define a scaling factor for all measurements regarding this input channel and also change the unit which is displayed throughout the tools and settings
Scale	numeric scaling factor	positive or negative real number to scale all measurement values attributed to that particular input; only appears when Scaling + Units is enabled
Units	three letter acronym	three letters can be set to define a unit, e.g. PSI; only appears when Scaling + Units is enabled
Range (unit)	1 mV to 2.1 V (or equivalent after unit + scaling is applied)	set the range of input 1/2 in fine increments within the available range - the value is the absolute range of the signal including a potential DC offset - note: the value inserted by the user may be approximated to the nearest value supported by the HF2; Scaling + Units is fully accounted for in this setting
A = Auto Range button	[press once]	press this button to automatically set the input range to two times the maximum amplitude of the input signal over a measured time of 100 ms
AC switch	OFF: DC coupling	set the input coupling for input 1/2 - an AC coupling inserts a high-pass filter at 1 kHz in the input path
	ON: AC coupling	
Differential switch	OFF: single-ended input	select whether to connect to a single-ended signal (one cable) or to a differential signal (two cables)
	ON: differential input	
50 Ω switch	OFF: 1 MΩ	set the matching impedance for input 1/2
	ON: 50 Ω	
Oscillator selection (1..6)	Osc1 to Osc6	select the oscillator for the related generator and demodulator
Frequency (Hz)	0 to 100 MHz	frequency for internal reference 1/2: this frequency can also be output with variable amplitude
Harm (1..6)	1 to 1023	selects the harmonic to be measured - enter 1 to measure the fundamental frequency and 2 for the 2nd harmonic, and so on
Phaseshift (1..6)	-180° to 180°	defines the phase shift of the oscillator associated to the demodulator and is relevant for the reference signal as well as for the signal output (works for internal and external reference modes)
Reference selection	Internal (internal reference)	internal reference mode for input 1/2: the reference is generated by internal oscillators
	Signal Input 1 (auto reference)	auto reference mode for input 1/2: the reference is derived from the input signal

Control/Tool	Options/Range	Description
	Signal Input 2 (external reference)	external reference mode for input 1/2: the reference is the signal attached to Input 2 connector
	Signal Aux In 1 (external reference)	external reference mode for input 1/2: the reference is the signal attached to Aux In 1 connector
	Signal Aux In 2 (external reference)	external reference mode for input 1/2: the reference is the signal attached to Aux In 2 connector
	Signal DIO 0 (external reference)	external reference mode for input 1/2: the reference is the signal attached to DIO 0 connector
	Signal DIO 1 (external reference)	external reference mode for input 1/2: the reference is the signal attached to DIO 1 connector
Lock flag	OFF / ON	this flag indicates when the demodulator is locked to the selected reference frequency - note: this flag is only meaningful when external and auto reference modes are selected
Auto Zero button |←	[press once]	this button shifts the phase of the reference at the input of the demodulator in order to achieve 0 phase at the demodulator output - this action maximizes the X output, zeros the Y output, zeros the Θ output, and leaves the R output unchanged
Signal Channel selection	1: the filter is connected to signal input 1	set the HF2 signal input that is connected to the related demodulator - the HF2 has 2 differential inputs that can be selected
	2: the filter is connected to signal input 2	
Filter Roll Off selection	1st: 6 dB/oct	set the filter roll off (filter order) for the related demodulator
	2nd: 12 dB/oct	
	3rd: 18 dB/oct	
	4th: 24 dB/oct	
	5th: 30 dB/oct	
	6th: 36 dB/oct	
	7th: 42 dB/oct	
	8th: 48 dB/oct	
Filter Property Unit selection	BW 3 dB: 80 µHz (filter order = 8) to 200 kHz (filter order = 1)	Filter properties can be set and displayed in either of these bandwidth or time constant units that differ by a scaling factor depending on the filter order - please refer to Table 10.1 for exact relation between TC, BW, and filter order - note: the value inserted by the user may be approximated to the nearest value supported by the HF2
	BW NEP: 90 µHz to 319 kHz	
	TC eff: 783 ns to 1900 s	

Control/Tool	Options/Range	Description
	TC per order: 783 ns to 580 s	
Sinc Filter switch (1..6)	OFF: Sinc filter disabled for corresponding demodulator	the Sinc filter is an additional filtering stage that permits to remove the omega and 2 times omega components - by using a large filter roll-off value, the user can effectively reduce the signal component at the second and third harmonics - for applications where large roll-off is not possible, the Sinc filter achieves the same effect - note: there are limitations regarding the maximum frequency and the frequency resolution allowed with Sinc filtering (see Table 4.2)
	ON: Sinc filter enabled for corresponding demodulator	
Block Lock button	[press once]	this button locks the settings between the different demodulators - when lock is closed, then changing one value changes all others - when lock is open, changing one leaves others unchanged
Trigger selection	Off	turn off demodulator
	Continuous	automatic internal trigger with the sample rate defined in the next control field
	DIO 0	samples are sent to the host computer depending on DIO 0 triggering
	DIO 1	samples are sent to the host computer depending on DIO 1 triggering
	DIO 0 or 1	samples are sent to the host computer depending on DIO 0 and 1 triggering
	Undefined	a complex trigger has been programmed to the HF2 Instrument by means of the programming interfaces - this mode cannot be selected by the user of ziControl, but is the result of a complex trigger definition - note: this is a read-only position
Trigger Mode selection	Rising Edge	1 data sample is sent to the host computer for a rising edge on the trigger input
	Falling Edge	1 data sample is sent to the host computer for a falling edge on the trigger input
	Both Edges	1 data sample is sent to the host computer for a rising edge and falling edge on the trigger input
	Gate High	data samples are sent to the host computer as long as the defined DIO 0/1 input is high - it is possible to determine the number of demodulated samples that are sent to the host computer after a trigger by setting the readout rate and modulating the pulse length on the DIO 0/1
	Gate Low	data samples are sent to the host computer as long as the defined DIO 0/1 input is low - it is possible to determine the number of demodulated samples that are sent to the host computer after a trigger by setting the readout

Control/Tool	Options/Range	Description
		rate and modulating the pulse length on the DIO 0/1
	Undefined	a complex trigger has been programmed to the HF2 Instrument by means of the programming interfaces - this mode cannot be selected by the user of ziControl, but is the result of a complex trigger definition - note: this is a read-only position
Sampling Rate (Sa/s)	0.22 Hz to 460 kHz	set the filter readout rate, equivalent to the demodulator samples (measured data points) that are sent to the host computer per second - it is also the rate of data received by ziControl and saved to the computer hard disk - this setting has no impact on the sample rate on the physical auxiliary outputs - note: the value inserted by the user may be approximated to the nearest value supported by the HF2 (see Readout Rate Overview)
Signal Output Amplitudes	0 to 10 V	set the output voltage for each individual demodulator for the two signal outputs1/2, the granularity depends on the signal output 1/2 range - note: the value inserted by the user may be approximated to the nearest value supported by the HF2
Range selector	±10 mV	set the output range for output 1/2 - this select determines the maximum output peak to peak range - this setting will make sure that no peaks above the setting are generated at the output, independent from the amplitude settings (V) - the output signal is clipped if the amplitude is higher than the range - if the range is changed to a value smaller than V, then V is automatically reduced to the new range
	±100 mV	
	±1 V	
	±10 V	
Amplitude indicator	0 to 100%	graphical indicator on how much of the set output range is currently used
Add switch	OFF: Add disabled	the Add input allows to add an external analog signal to the internally generated signal - when disabled, the signal at the Add input is ignored
	ON: Add enabled	
Output switch	OFF: output disabled	set the main switch for output 1/2 - when the output is enabled, then the HF2 signal output is active and the blue LED on the front panel lights-up
	ON: output enabled	

4.2.3. Modulation Settings Tab

Note

The modulation tab appears only when the HF2LI-MOD option has been purchased and activated. Customers can purchase the HF2LI-MOD modulation option at any time, whether when ordering their instrument or after delivery. This option can be activated by the user or by Zurich Instruments via remote servicing.

Note

The HF2LI-MOD modulation option requires that the HF2LI-MF multi-frequency option to be purchased and activated.

Features:

- Control for AM and FM narrow-band demodulation
- Control for AM and FM generation
- Direct sideband analysis

Description:

For AM generation, three demodulators from the multi-frequency option are used to produce the carrier and the two sidebands (if both sidebands are selected). Two separate modulation units are provided, Mod1 unit activates Demodulators 1, 2 and 3, and using the Mod2 unit activates Demodulators 4, 5 and 6. Carrier and sideband amplitudes can be selected in the Output Amplitudes menu and the signal can be generated on Signal Output 1, Signal Output 2 or on both: selecting the output on the pull-down menu then selects the correct amplitudes on the Lock-in MF tab and activates them.

The demodulation of an AM signal also requires three demodulators as the Lock-in MF tab will show. If single sideband modulation/demodulation is selected, one demodulator less is required. In the Numerical tab, the Demodulator 2 (5 if Mod2 is used) is active and 3 (respectively 6) is grayed out.

One should not confuse oscillators with demodulators. In principle all oscillators and demodulators can be used for modulation schemes.

For FM generation, two demodulators are used (either 1,2 or 4,5) to generate the carrier frequencies and the sidebands according to the defined Carrier (V) amplitude setting and Modulation Index. FM detection on the other hand always requires 3 demodulators. By choosing higher harmonics for sideband demodulators one can select higher order sidebands.

Note

The resulting frequencies f1+f2 and f1-f2 are only available inside the instrument, and not displayed nor stored by ziControl. Therefore in the numerical tool, in the spectroscope tool, or in the samples saved on the hard disk, the modulation frequency is indicated instead of the true demodulation frequency.

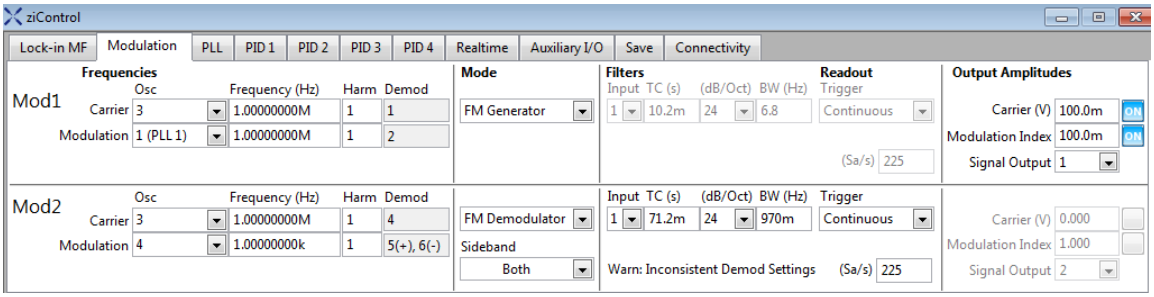


Figure 4.4. Modulation settings tab

Table 4.4. Modulation settings tab

Control/Tool	Options/Range	Description
Carrier Osc selection	1 (PLL 1), 2 (PLL 2), 3, 4, 5, 6	select the oscillator (or PLL) to be used as the carrier signal
Carrier Frequency (Hz)	0.8 µHz to 100 MHz	set the carrier signal frequency - note: the value inserted by the user may be approximated to the nearest value supported by the HF2
Frequency Harm	1 to 1023	set the harmonic of the carrier signal
Modulation Osc selection	1 (PLL 1), 2, 3, 4 (PLL 2), 5, 6	select the oscillator (or PLL) to be used as the modulation signal
Modulation Frequency	0.8 µHz to 100 MHz	set the modulation signal frequency - note: the value inserted by the user may be approximated to the nearest value supported by the HF2
Modulation Harm	1 to 1023	set the harmonic of the modulation signal
Carrier and Modulation Demod	read only	field provides information on which demodulators are currently allocated for the mode defined with the following setting
Modulation Mode selection	Off	no action
	AM Gen+Demod	MOD operates in amplitude generation and demodulation mode
	FM Generator	MOD operates in FM generator (modulation) mode
	FM Demodulator	MOD operates in FM demodulation mode
Sideband selection	Both	MOD operates in dual-sideband mode - note: setting is grayed-out when not applicable
	Single-Sided +	MOD operates in single-sideband (positive) mode - note: setting is grayed-out when not applicable
	Single-Sided -	MOD operates in single-sideband (negative) mode - note: setting is grayed-out when not applicable
Input Signal selection	1	demodulation is performed on Signal Input 1 - note: setting is grayed-out when not applicable
	2	demodulation is performed on Signal Input 2 - note: setting is grayed-out when not applicable
Time Constant, Filter Roll Off, Bandwidth	0.8 µs to 580 s, 6 to 48 dB/Oct, 80 µHz to 200 kHz	set the filter properties of all modulation relevant filters simultaneously (see also Table 4.1)

Control/Tool	Options/Range	Description
Warn: Inconsistent Demod settings	message	indicates whenever the 2 or 3 modulation related demodulators have different properties set
Trigger	Off, Continuous, DIO 0, DIO 1	see Table 4.1 for more details on trigger settings)
Carrier Signal Amplitude	8 µV to 10 V	set the amplitude of the carrier signal - note: the maximum value is limited by the range setting on the Lock-in tab
Modulation Amplitude (AM)/ Modulation Index (FM)	8 µV to 10 V	set the amplitude of the modulation signal - note: the maximum value is limited by the range setting on the Lock-in tab - note: setting is grayed-out when not applicable
Signal Output selection	1	generated signal is available on Signal Output 1
	2	generated signal is available on Signal Output 2
	1&2	generated signal is available on both, Signal Output 1 and Signal Output 2

4.2.4. PLL Settings Tab

Note

The PLL tab appears only when the HF2LI-PLL option has been purchased and activated. Customers can purchase the HF2LI-PLL phased-locked loop option at any time, whether when ordering their instrument or after delivery. This option can be activated by the user or by Zurich Instruments via remote servicing.

The HF2LI-PLL feature provides 2 independent PLL for high speed, high accuracy, highly flexible tracking of frequency modulated signals. The PLL permits to recover such signals with a powerful user interface with the choice of many automatic and manual settings.

Features:

- Dual fully programmable 50 MHz phased-locked loops
- Programmable PLL center frequency and set point
- Programmable PLL phase detector filter settings and PI controller dynamics
- Comfortable PLL Advisor panel for transfer function analysis
- Advanced 2-omega PLL mode (requires access to HF2LI-MF option)
- Auto-zero functions for center frequency and set point

Description:

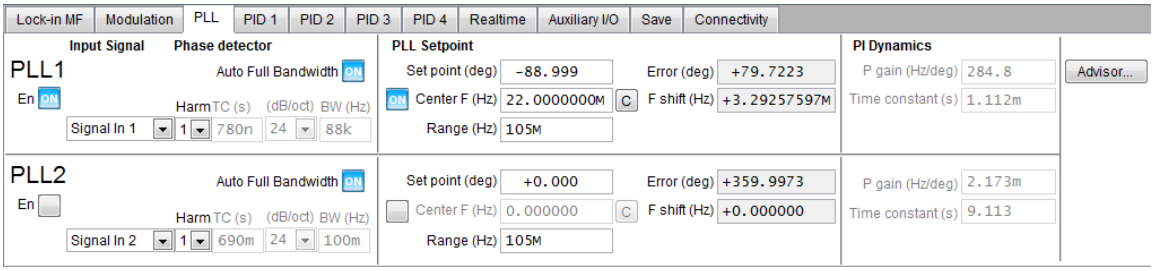


Figure 4.5. PLL Settings Tab

Table 4.5. PLL settings tab

Control/Tool	Options/Range	Description
PLL 1/2 enable	OFF: PLL disabled	disables the PLL - the related oscillators run at the frequency programmed in the corresponding registers
	ON: PLL enabled	enables the PLL - the related oscillators run controlled by the PLL
PLL 1/2 signal input select	Signal In 1	select the source for the reference signal of the PLL - this is the signal the PLL will lock to - the PLL can take a reference signal from either the high-speed analog inputs on the front-panel or the digital I/O and auxiliary inputs on the rear-panel
	Signal In 2	
	Aux In 1	
	Aux In 2	
	DIO 0	
	DIO 1	
PLL 1/2 phase detection mode	1 (normal PLL mode)	performs the phase detection on the first harmonic
	2 (2-omega PLL mode)	performs the phase detection on the second harmonic of the PLL drive frequency - with 2-omega mode the PLL lock is performed on the physical unit that takes place 2 times faster than the drive frequency - note: this advanced mode should only be used with manual center frequency and range setting - otherwise there is a chance to lock on the wrong frequency
PLL 1/2 phase detector mode	OFF: automatic full bandwidth	the phase detector low pass filter and the PI controller is defined with automatic settings derived from the PLL center frequency settings
	ON: manual mode	the phase detector low pass filter and the PI controller can be controlled with manual settings
PLL 1/2 phase detector time constant (TC)	0.8 µs to 580 s	sets the time constant of the phase detector in the PLL - note: the value in this field is related to the value entered in the BW field
PLL 1/2 phase detectors filter order	1st: 6 dB/oct	sets the filter order for the phase-detector of the PLL
	2nd: 12 dB/oct	
	3rd: 18 dB/oct	
	4th: 24 dB/oct	
	5th: 30 dB/oct	
	6th: 36 dB/oct	

Control/Tool	Options/Range	Description
	7th: 42 dB/oct	
	8th: 48 dB/oct	
PLL 1/2 phase detector bandwidth (BW)	80 µHz (filter order = 8) to 200 kHz (filter order = 1)	sets the filter bandwidth of the phase-detector of the PLL - note: the value in this field is related to the value entered in the TC field
PLL 1/2 set point field	-180° to 180°	defines the set point of the PLL - the PLL will adjust the frequency of its oscillator to have this phase difference compared to the reference signal - the value can be copied from the current phase information from the demodulator 1/2 by pressing the button marked "C"
PLL 1/2 set point copy button	[press once]	updates the set point of the PLL by performing an auto zero of the phase error
PLL 1/2 phase error	-	indicates the phase error at the input of the PLL phase detector - if the PLL is in locked state, the phase error is very small (much smaller than 1 degree)
PLL 1/2 center frequency mode	OFF: automatic mode	the PLL center frequency is determined automatically - in this mode the instrument sweeps the operating range until it finds a suitable frequency - note: automatic center frequency mode only works for open loop systems; closed loop systems requires manual mode
	ON: manual mode	the PLL center frequency is determined manually by the field on the right
PLL 1/2 center frequency field	1 Hz to 50 MHz	defines the center frequency of the PLL - the actual PLL frequency can be copied to this field by pressing the button marked "C" - note: in order the copy to work, the PLL must be reasonably locked
PLL 1/2 center frequency copy button	[press once]	updates the center frequency of the PLL by performing an auto zero of the frequency shift
PLL 1/2 frequency shift	-	indicates the frequency shift between the actual PLL frequency and the PLL center frequency f_{PLL} - f_{CENTER} - for example, if the center frequency is 1 MHz and the reference signal has a frequency of 1.001 MHz, the frequency shift is -1 kHz
PLL 1/2 frequency shift range	1 Hz to 100 MHz	limits the maximum frequency shift and therefore the minimum and maximum excitation frequency (PLL output) - this settings has the capability to prevent resonator damage by applying frequencies in the wrong range
PLL 1/2 PI proportional gain	5 mHz/deg to 140 kHz/deg	sets the proportional gain of the PI controller in the PLL - when disabled, the proportional gain is determined automatically based on the center frequency
PID 1/2 PI time constant	180 ns to 100 ks	set the integral time constant of the PI controller in the PLL - when disabled, the integral time

Control/Tool	Options/Range	Description
		constant is determined automatically based on the center frequency
Advisor	[press once]	invokes the PLL Advisor window to estimate the system transfer function

Note

The proportional gain G of the PI controller of the PLLs is too large by a factor of 2. This means that if it reads 100, the correct value should be 50. Since K_P = G and K_I = G / TC, the integral gain will scale in the correct way after correcting G.

The PLL Advisor supports the user estimate the system transfer function consisting of PLL and external resonator. After the transfer function analysis, the user can feed the PLL with the settings calculated by the PLL Advisor and thus expect a predictable system behavior. Experience has shown that the PLL Advisor and the real PLL behave similarly as long as there is no overshoot in the transfer function.

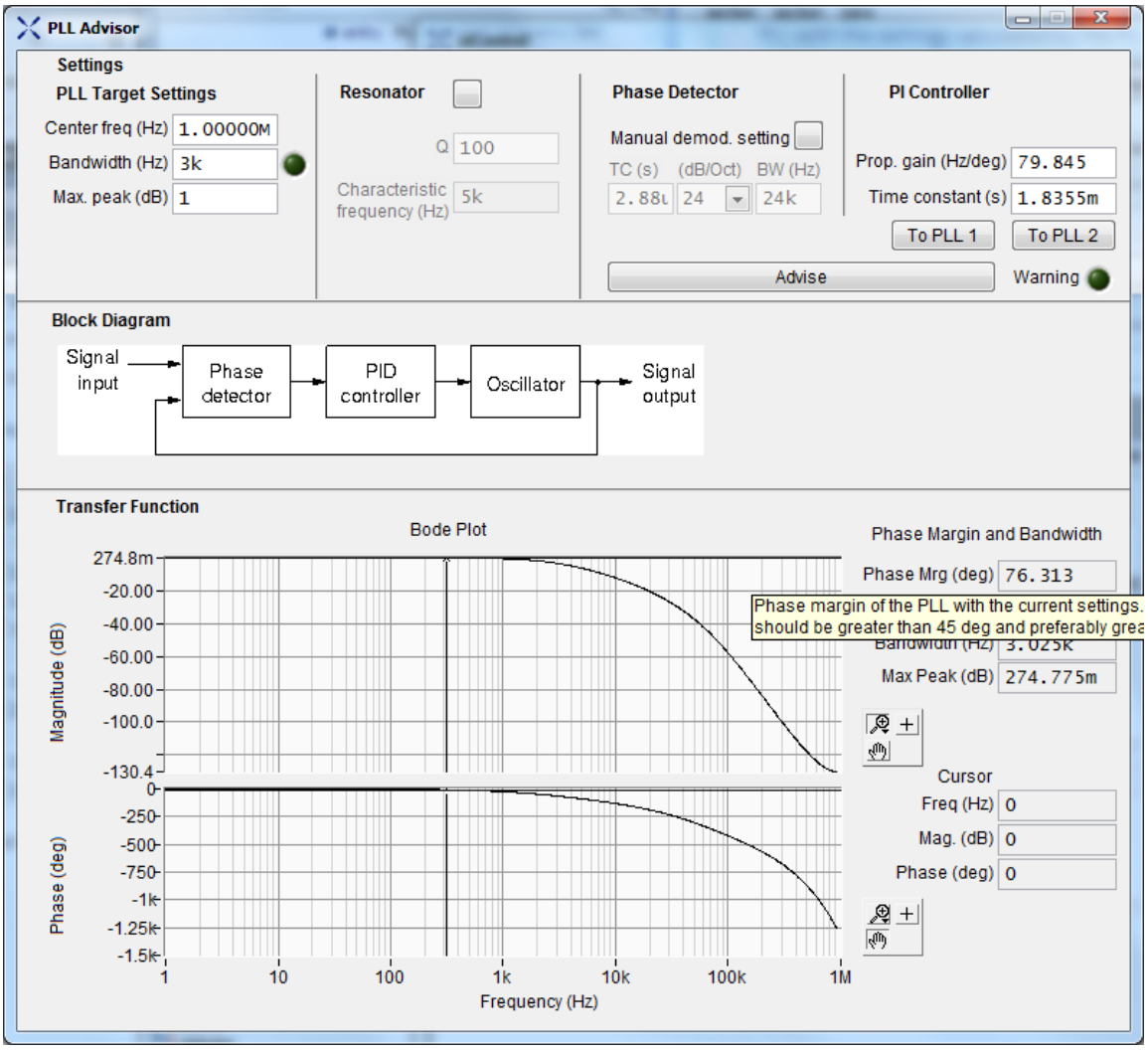


Figure 4.6. PLL Advisor

Table 4.6. PLL Advisor

Control/Tool	Options/Range	Description
Target center frequency	1 Hz to 50 MHz	defines the center frequency for the parameter estimation
Target bandwidth	16 Hz to 100 kHz	defines the measurement bandwidth for the parameter estimation
Target maximum overshoot	1 mdB to open	defines the maximum overshoot that is tolerated by the resonator
Resonator switch	OFF	the transfer function of the system is calculated using an internal look-up table simulating a response dependent solely on the center frequency (also ignores the target bandwidth)
	ON	a resonator with below mentioned characteristics is used for the estimation
Resonator Q factor	1 m to 1 T	defines the Q factor of the external resonator - the Q factor characterizes the bandwidth of the resonator relative to the center frequency $Q = f_R / BW$ - a higher quality factor indicates a lower energy loss compared to the stored energy in the resonator, which implies an oscillation with higher amplitudes, but a smaller range around the center frequency in which the oscillation is possible
Resonator characteristic frequency	-	indicates the characteristic frequency of the resonator $f_R = f_C / (2 * Q)$, representing the bandwidth of the resonator
PLL phase detector setting mode	OFF: automatic mode	calculates the phase detector settings based on the center frequency setting
	ON: manual mode	permits the user to define the own settings for the phase detector filter, and use the PLL Advisor to calculate the PID parameters
PLL phase detector filter TC	0.8 µs to 580 s	sets the filter time constant of the phase detector for estimation
PLL phase detector filter order	1st to 8th	sets the filter order of the phase detector for estimation
PLL phase detector filter BW	80 µHz (filter order = 8) to 200 kHz (filter order = 1)	sets the filter bandwidth of the phase detector for estimation
PID proportional gain	10 mHz/deg to 10 kHz/deg	PID proportional characteristic for the transfer function calculation / estimation
PID integral time constant	100 µs to 1s	PID integral characteristic for the transfer function calculation / estimation
Advise button	[press once]	the PLL Advisor calculates the system transfer function based on the settings (see Bode plot), and calculates settings for the phase detector (if manual) and
Warning flag	-	the PLL Advisor was not able to calculate a satisfying solution - thus cross-check the result (example: bad overshoot, small phase margin)

Control/Tool	Options/Range	Description
To PLL 1 button	[press once]	copies the estimated phase detector and PID controller settings as active settings for PLL 1 - note: the settings have immediate effect
To PLL 2 button	[press once]	copies the estimated phase detector and PID controller settings as active settings for PLL 2 - note: the settings have immediate effect
Result phase margin	-	indicates the phase margin - a reasonable phase margin target is 65 degrees, though most systems will also operate well with 45 degrees
Result phase margin frequency	-	indicates the frequency where the phase margin is measured
Result bandwidth	-	indicates the 3 dB point of the closed loop transfer function
Result maximum peak	-	indicates the simulated maximum overshoot in dB
Transfer function Bode plot	-	represents the transfer function as estimated with the above settings of the resonator (if selected), the phase detector, and the PID controller - the Bode plot consists in magnitude (in dB) and phase (in degrees) of the transfer function of the complete system - the meaning of the magnitude relates how frequency modifications are transferred from the input to the output of the PLL
Bode plot cursor values	-	indicate the magnitude, phase, and frequency of the current cursor position

Note

The interpretation of the Bode plot is not within the scope of this document. Two comments regarding Bode plots are still to be mentioned: the overall stability of the system can be determined by the amount of overshoot that the system allows and by the phase margin the system has at the system bandwidth (3 dB point).

4.2.5. PID Settings Tab

Note

This tab appears only when the HF2LI-PID option has been purchased and activated. Customers can purchase this option at any time, whether when ordering their instrument or after delivery. It can be activated by the user or by Zurich Instruments via remote servicing. Some of the settings depend on the availability of other options.

Note

The HF2LI-PID option cannot be used at the same time as the HF2LI-RT option. If the PID option is installed on your instrument, then it is available after power-up. After downloading a RT program into the HF2 Instrument, the PID option is not available anymore. In order to re-activate the PID controller the user needs to either restart ziServer or to power cycle their HF2 Instrument.

Note

The modifications of some settings of the HF2LI-PID option modify the settings in other panels/ options.

Features:

- Four independent proportional, integral, derivative (PID) controllers
- Automatic P, I, and D parameter tuning for different system models (DUT)
- Bode plots to aid PID parameter tuning
- Simulated step response plots for the system model (DUT) and closed loop
- Output center and range setup (anti-windup)
- Eleven selectable input units for each controller
- Four selectable sources for the set point
- Four selectable output units for each controller
- Output to auxiliary output connectors
- Arithmetic processor for specific calculations
- Default output value for disabled controllers

Description:

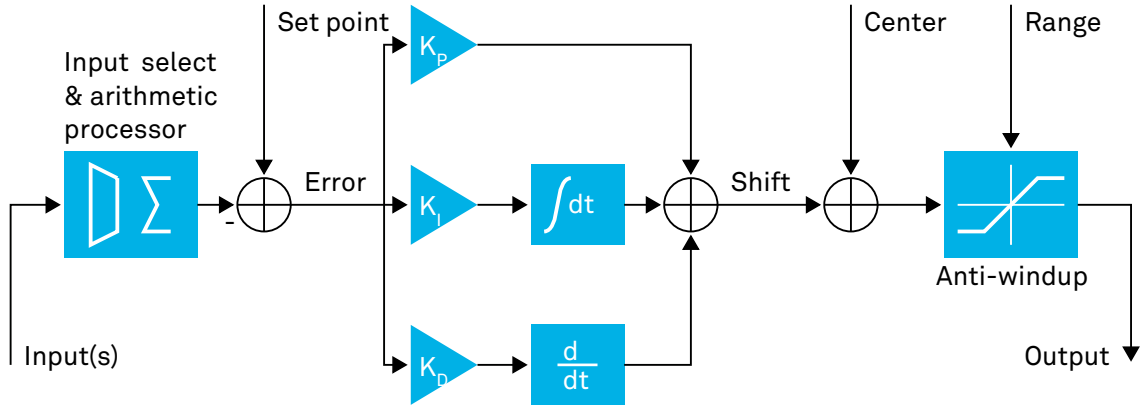


Figure 4.7. HF2LI-PID block diagram

The HF2LI-PID option serves to post-process a wide selection of input units in order to control a selected output by means of a proportional, integral, derivative controller. Each PID controller strives to reduce the Error that results from the subtraction of the input unit from the Set point. The Shift signal is then calculated according to the formula below and added to the Center. Before the output is generated, the anti-windup logic circuits makes sure that the Range is not exceeded at any time preventing illegal output ranges that could harm the following circuits.

$$\text{Shift} = K_P * \text{Error} + K_I * \int \text{Error} dt + K_D * \frac{d}{dt} \text{Error}$$

Equation 4.1. PID controller generic equation

The P, I, and D parameters are mapped as follows K_P = P, K_I = I, K_D = D.

The PID settings tab is divided into several sections labeled Input, Output, Control and DUT System Model. On the top right corner there is a rate indicator that provides an instant information of the PID output rate for all controllers. The rate decreases when more PID controllers are enabled, indicating that the same rate holds for all controllers. Although the bandwidth of a PID controller must be measured in action, it is possible to empirically calculate the bandwidth by dividing the output rate by 10, which is a conservative value.

Input:

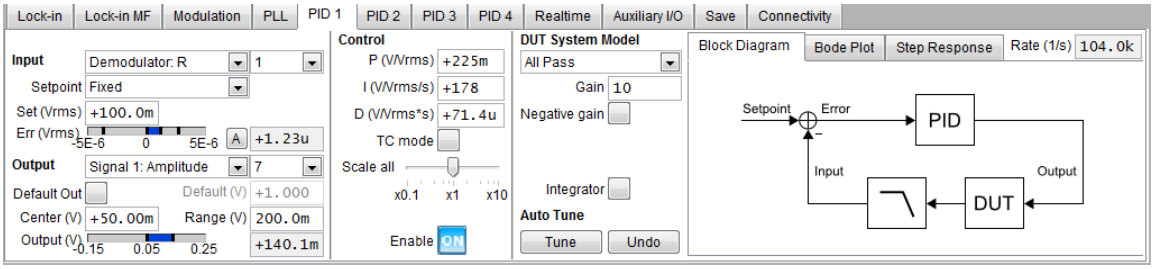


Figure 4.8. PID settings tab

Table 4.7. PID input settings

Control/Tool	Options/Range	Description
Input selection	Demodulator X, 1-6	defines the input unit for the PID controller as the output of a demodulator - for physical units see Table 4.12 .
	Demodulator Y, 1-6	
	Demodulator R, 1-6	
	Demodulator Θ, 1-6	
	Aux Input, 1-2	defines the input unit for the PID controller as the auxiliary inputs or the feedback value of the auxiliary outputs - for physical units see Table 4.12 .
	Aux Output (used as input), 1-4	
	AM Index, 1-2	defines the input unit to be the modulation index from an amplitude modulated signal (only available if the HF2LI-MOD option is available and active)
	Demodulator $\lvert Z_{i+1} \rvert$ - $\lvert Z_i \rvert$, 1-5	uses the arithmetic processor to combine the outputs of two consecutive demodulators, where Z_i is the complex signal, $Z_i = X_i + iY_i$ composed by the X_i and Y_i output of demodulator i
	Demodulator $\lvert X_{i+1} \rvert$ - $\lvert X_i \rvert$, 1-5	
	Demodulator $\lvert Z_{i+1} - Z_i \rvert$, 1-5	
	Oscillator Frequency, 1-6	defines the input unit to be an oscillator frequency - note: the precision of the frequency is limited to 32-bit floating-point precision (~8 digits of precision) - e.g. at 500 Hz the resolution is approximately 20 mHz
Set point selection	Fixed	manual PID mode, sets a constant set point

Control/Tool	Options/Range	Description
	Toggle	toggle PID mode, toggles the PID set point between two fixed values at a rate of 1 Hz - this toggle mode is excellent to tune the dynamics of the controller settings
	Aux Input, 1-2	external PID mode, reads the set point from the auxiliary inputs
	PID N-1	cascaded PID mode, sets the set point of controller PID(n) using the output of the previous controller PID(n-1), more precisely PID(mod(n +2,4)+1))
Set	depends on input selection	defines a fixed set point - depending on the previous setting, there might be one, two, or no fields at all
Error range	-	displays the calculated PID error (Error = Set point - Input) as graphical range, it is possible to manually change the range by clicking onto the label inside ziControl
Auto button	[press once]	automatically changes the range of the graphical range display - does not change any PID setting
Error value	-	displays the calculated PID error (Error = Set point - Input) as numerical value

The three input units that make use of the arithmetic processor are explained hereafter. In fact the outputs of two consecutive demodulators are used to compute the unit that is used as PID controller input and subtracted from the set point. The available configurations serve as follows:

- Demodulator $|Z_{i+1}|$ - $|Z_i|$: dual-frequency mode, and is equivalent to the notation R_{i+1} - R_i
- Demodulator X_{i+1} - X_i: Kelvin probe force microscopy
- Demodulator $|Z_{i+1}$ - $Z_i|$: AM and FM sideband analyzer mode

When selecting one of these three input units, the selection range changes to a dual-digit notation, e.g. (1,2) meaning demodulator 1 and 2. Note the difference between the first and the third mode two PID inputs: The first subtracts the magnitude of demodulator i+1 from demodulator i, whereas the second subtracts the complex signals as a vector before calculating the magnitude of the result.

It is not the objective of this section to explain the purpose and application of these three input units. The Zurich Instruments blogs explain these modes instead.

Output:

Table 4.8. PID output settings

Control/Tool	Options/Range	Description
Output selection	Signal 1 amplitude, oscillator 1-8	defines the output unit of the PID controller to regulate the amplitude of Signal Output 1 - the related oscillator needs to be considered as well - for physical units, see Table 4.12
	Signal 2 amplitude, oscillator 1-8	defines the output unit of the PID controller to regulate the amplitude of Signal Output 2 - the related oscillator needs to be considered as well - for physical units, see Table 4.12

Control/Tool	Options/Range	Description
	Oscillator frequency, 1-6	defines the output unit of the PID controller to regulate the frequency of the defined oscillator - note: the precision of the frequency is limited to 32-bit floating-point precision (~8 digits of precision), e.g., at 500 Hz the resolution is approximately 20 mHz
	Aux output offset, 1-4	defines the output unit of the PID controller to regulate the offset setting of the defined auxiliary output
	DIO (digital 16 bit)	defines the output unit of the PID controller to regulate the DIO parallel port encode in int16 format
Default Out enable	OFF	the PID output is the last driven value when the PID controller is disabled
	ON	permits to define the default PID output when the PID controller is disabled
Default	-∞ to +∞	the default value that is written to the output if the PID controller is disabled and Default Out is enabled
Center	depends on input selection	adds an offset to the PID output
Range	0 to +∞	defines the range of the PID output to prevent windup or to protect external hardware - the absolute limits of the output are [Center - Range, Center + Range] - the Range can be set to have negative output values - beware that in combination with some input units (e.g. negative values of R), this might have unexpected behavior
Output range	-	displays the PID output as graphical range - the calculated value is Output = Center + Shift, if the PID output reaches either the user-specified limits or the instruments hardware limits, the indicator will turn red
Output value	-	displays the PID output (Output = Center + Shift) as numerical value

Control:

Table 4.9. PID parameter settings

Control/Tool	Options/Range	Description
P (proportional gain)	-∞ to +∞	sets the proportional gain for the error signal K_P = P
I (integral gain)	-∞ to +∞	sets the integral gain for the error signal K_I = I
D (derivative gain)	-∞ to +∞	sets the derivative gain for the error signal K_D = D
TC mode control	OFF: PID mode	enables the determination of the gain parameters by the PID constants
	ON: TC mode	enables the determination of the gain parameters by the proportional gain K_P = P and the time constants T_I = P/I and T_D=D/P

Control/Tool	Options/Range	Description
Scale all slide bar	0.1 to 10	scales all gain parameters of the PID by a scalar factor between 0.1 and 10
Enable control	OFF	disables the PID controller - the input signals are tracked and the error signal is calculated, but no control loop is applied
	ON	enables the PID controller - the input signals are tracked, the error is calculated and the control loop is applied

DUT System Model:

All the settings in this section serve to define an efficient DUT model to be simulated together with the required bandwidth in order to determine optimal PID parameters. There is also a block diagram that shows the simulated units, and a frequency domain as well as a time domain analysis.

The block diagram serves to highlight the blocks that are simulated and displayed in the diagrams. This diagram actually depends on several settings in the PID tab, but also from settings in other tabs. The PID parameters are the result from the simulation and are read in the Control section. It is possible to modify the parameters and to immediately see the effects in the bode diagram and in the step response. The DUT model is defined with the settings from the table below. When a demodulator is selected as PID input, then the block diagram will display a low-pass filter which corresponds to the settings of the demodulator filters.

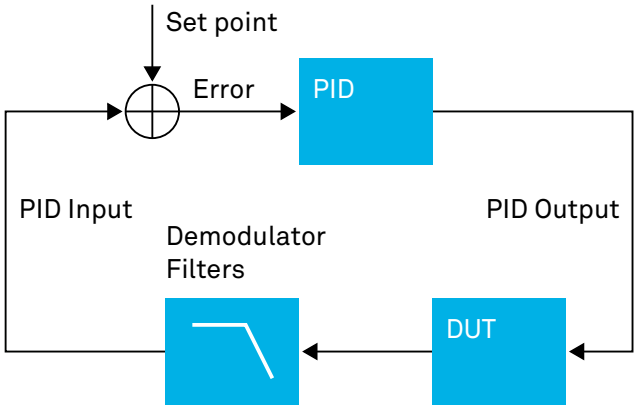


Figure 4.9. PID system diagram

Table 4.10. PID DUT model settings

Control/Tool	Options/Range	Description
DUT System Model selection	All Pass	models the DUT using an all pass filter, e.g., a feedback loop cable
	1st Order Low Pass	models the DUT using a first order low pass filter
	2nd Order Low Pass	models the DUT using a second order low pass filter
	AFM Cantilever Amplitude Control	models via AFM cantilever amplitude control
DUT System Model parameters	Gain (all models)	defines the gain applied in the plant models, positive gain parameters correspond to positive feedback, negative gain parameters correspond to negative feedback

Control/Tool	Options/Range	Description
	Negative Gain (all models)	inverts the gain defined in the Gain field (only applies to the resulting auto tune gains)
	BW (1st and 2nd order)	defines the characteristic bandwidth to use in the model [Hz]
	Damping (2nd order)	defines the damping used in the 2nd Order Low Pass model, with damping = 1/(2Q) and Q being the Q-factor - a critically damped system corresponds to damping of 1, or Q = 1/2
	Q (AFM cantilever)	defines the Q-factor used in the AFM Cantilever model
	F (AFM cantilever)	defines the frequency of the AFM cantilever used in the AFM cantilever model [Hz]
Integrator switch	OFF	does not use an additional integrator in the System (DUT) Model
	ON	uses an additional integrator in the System (DUT) Model - example use of this feature taking demodulator phase as PID input and driving oscillator frequency as PID output
Tune button	[press once]	calculates and applies optimized PID gain parameters for the PID controller
Undo button	[press once]	undoes the last Tune to the parameters used before pressing Tune

Table 4.11. PID system model simulation

Control/Tool	Options/Range	Description
Bode Plot tab	System (DUT)	plots the transfer function of the defined system DUT
	Demod Filter	plots the demodulator filter transfer function, when the PID input is selected to a demodulator output - this setting is grayed out when something else than a demodulator is selected
	System & Filter	plots the transfer function of the system DUT and the filter
	Open Loop	plots the open loop transfer function and makes an analysis of the gain margin (GM) and the phase margin (PM)
	Closed Loop	plots the closed loop transfer function and determines the effective system bandwidth and warns if there is a closed loop stability problem
Step Response tab	System (DUT)	plots the step response of the defined system DUT
	Closed Loop	plots the step response of the closed loop

Table 4.12. PID control units

Class	Signal	Unit
Input unit	Demodulator X value	$[\text{Input unit}]_{\text{RMS}}$

Class	Signal	Unit
	Demodulator Y value	[Input unit]$_{RMS}$
	Demodulator R value	[Input unit]$_{RMS}$
	Demodulator Θ value	°
	Auxiliary input	V
	Auxiliary output (used as input)	V
	Modulation index	In the range [0,1]
	Demodulator: $X_{n+1} - X_n$	[Input unit]$_{RMS}$
	Demodulator: $\lvert\vec{z}_{n+1}\rvert - \lvert\vec{z}_n\rvert$	[Input unit]$_{RMS}$
	Demodulator: $\lvert\vec{z}_{n+1} - \vec{z}_n\rvert$	[Input unit]$_{RMS}$
	Oscillator frequency	Hz
Output unit	Signal output amplitude	V_{RMS}
	Oscillator frequency	Hz
	Auxiliary output	V
	DIO	TTL, encoded int16 format, referenced to least significant bit (LSB)
Control gain parameters	P	[Output unit] / [Input unit]
	I	[Output unit] / [Input unit] / s
	D	[Output unit] / [Input unit] * s
Control settings	Set point	[Input unit]
	Center	[Output unit]
	Range	[Output unit]
Calculated control parameters	Output	[Output unit]
	Error	[Input unit]

4.3. Other Settings

4.3.1. Real-time Settings Tab

Note

The real-time tab appears only when the HF2LI-RT / HF2IS-RT option has been purchased and activated. Customers can purchase the RT option at any time, whether when ordering their instrument or after delivery. This option can be activated by the user or by Zurich Instruments via remote servicing.

Features:

- Push button loading of real-time programs via the graphical user interface.
- Easy access to the embedded microprocessor's user registers to configure program parameters or view program output at run-time. Optional naming of user registers for easy identification, saving and opening of register names for persistent configuration across sessions.
- Displays real-time debugging information in the message log field box and/or allows the user to save the message log to file.
- Graphical CPU load indicator.

Description:

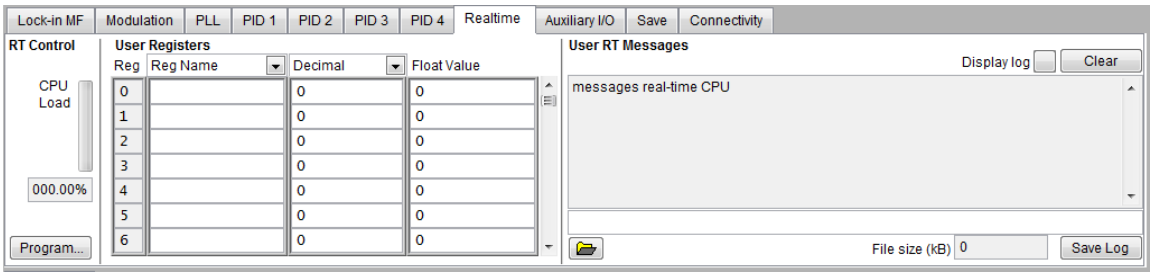


Figure 4.10. Real-time settings tab

Table 4.13. Real-time settings tab

Control/Tool	Options/Range	Description
CPU Load indicator	0% to 100%	indicates the momentary load of the embedded microprocessor running the real-time code
Program... push button	[press once]	selects a binary program to download to the HF2 Instrument; execution starts immediately thereafter
Reg Name pull-down menu	Reg Name	enters a name for the user register
	Save...	saves the user register names in an XML user register configuration file
	Open...	loads user register names saved in an XML user register configuration file
	Clear	clears entered user names. The values of the user register remain unaffected

Control/Tool	Options/Range	Description
Format pull-down menu	Decimal	selects the display format of the user register between decimal and hexadecimal
	Hexadecimal	
Float Value	IEEE754 32-bit floating-point number	view or edit a user register value as a 32-bit floating point number
Register fields 0-63	Register name & value	visualizes the processor register content - these fields are updated as the processor modifies the values - these fields are read/write: they permit to directly write values into the registers of the embedded processor. Each register can be optionally named for easy identification
User RT Messages	Read only	displays log messages printed via `ziRTKPrintf()` from the real-time program running on the embedded processor.
Display log toggle button	OFF	update the message log field disabled
	ON	update the message log field enabled
Clear push button	[press once]	clears the message log field
Log file name edit box	File name	specifies a destination file to save real-time log messages
Folder push button	[press once]	opens a dialog to specify a destination file to save real-time log messages
File size (kB)	Read only	displays the current size of the saved log file
Save log push button	[press once]	saves the real-time log messages in the specified file

4.3.2. Auxiliary I/O Settings Tab

Features:

- Control for auxiliary output connectors
- Monitor of auxiliary input connectors
- Monitor and control of digital I/O connectors

Description:

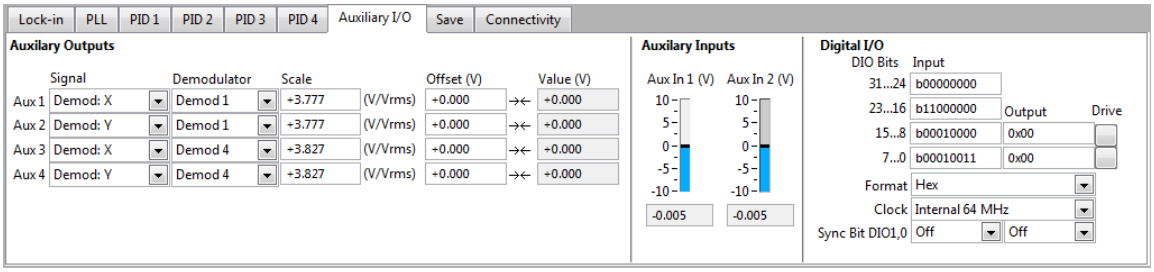


Figure 4.11. Auxiliary I/O settings tab

Table 4.14. Auxiliary I/O settings tab

Control/Tool	Options/Range	Description
Aux 1/2/3/4 output signal select	Manual (V)	set the signal that is provided on the Aux 1/2/3/4 connectors on the front panel of the instrument - the manual setting generates a DC value - the PLL:df selection is only available with the HF2LI-PLL option - the PID:Output selection is only available with the HF2LI-PID option
	X (V_{RMS})	
	Y (V_{RMS})	
	R (V_{RMS})	
	Theta (deg)	
	PLL1: dF (Hz)	
	PLL2: dF (Hz)	
	PID1: Output (PID output unit)	
	PID2: Output (PID output unit)	
	PID3: Output (PID output unit)	
	PID4: Output (PID output unit)	
Aux 1/2/3/4 output demodulator select	Demodulators	set the demodulator unit whose samples are being output to the related Aux output connector on the front panel of the instrument - note: this field is not available when the signal select is on manual or on PLL:dF or on PID:Output
Aux 1/2/3/4 output signal scale	-80G to 80G (V/V_{RMS}) for X , Y, R	set the scaling factor applied to the selection, consisting of a digital multiplication in real-time - the range of this field depends on the selected input range (see the following table) - note: this field is not available when the signal select is on manual
	-460k to 460k (V/deg) for Θ	
	-1.6 to 1.6 (V/Hz) for PLL:dF	
	-10^{37} to 10^{37} for PID:output	
Aux 1/2/3/4 output offset	-10 V to 10 V	set the DC offset value that is being added after the scaling of the selection
Aux 1/2/3/4 output value	[display]	instantaneous value at the corresponding auxiliary output connector
Aux In 1	-10 V to 10 V	indicates the current signal level on auxiliary input 1 (back panel)
Aux In 2	-10 V to 10 V	indicates the current signal level on auxiliary input 2 (back panel)
Digital I/O input bits 31...24	[input]	digital value in DIO input register bits 31 to 24
Digital I/O input bits 23...16	[input]	digital value in DIO input register bits 23 to 16
Digital I/O input bits 15...8	[input]	digital value in DIO input register bits 15 to 8
Digital I/O input bits 7...0	[input]	digital value in DIO input register bits 7 to 0

Control/Tool	Options/Range	Description
Digital I/O output bits 15...8	0x00 to 0xFF	digital value in DIO output register bits 15 to 8
Digital I/O output bits 7...0	0x00 to 0xFF	digital value in DIO output register bits 7 to 0
Digital I/O drive selector bits 15 to 8	OFF: no drive (input)	digital value at the DIO port bits 15 to 8 is not driven, high impedance, and the connectors can be used as inputs
	ON: drive (output)	digital value at the DIO port bits 15 to 8 is driven
Digital I/O drive selector bits 7 to 0	OFF: no drive (input)	digital value at the DIO port bits 7 to 0 is not driven, high impedance, and the connectors can be used as inputs
	ON: drive (output)	digital value at the DIO port bits 7 to 0 is driven
Digital I/O Hex/Bin	Hex	change between hexadecimal and binary representation
	Binary	
Digital I/O Clock	Internal 64 MHz	DIO input register is latched by internal 64 MHz clock
	Clk Pin 68	DIO input register is latched by signal on Pin 68 of the DIO port
Sync Bit DIO1,0	Off	DIO outputs 1 and 0 (BNC connectors) are not used for sync output and are free for other purposes
	Demod 1 to 8	reference signal of the selected demodulator is output on DIO 1 and 0 - note: there is a 166 ns delay between the sync and the front panel outputs (sync comes first) which leads to a relevant phase shift at high frequencies. To get an in phase sync, please make use of the WAVEFORM node setting and one of the HF front panel outputs

Table 4.15. Auxiliary output ranges

Input range setting	Scaling range for R (V/V_{RMS})	Scaling range for X and Y (V/V_{RMS})	Desired input full scale sensitivity (10 V full range output)
1 mV	80G	75G	1 nV to 1.5 V
10 mV	8G	7.5G	10 nV to 1.5 V
100 mV	760M	750M	100 nV to 1.5 V
1 V	83M	71M	1 µV to 1.5 V
2.1 V	42M	35M	2 µV to 1.5V

4.3.3. Save Settings Tab

Features:

- Save of demodulated samples to host computer disk drive
- Load and save of instrument configurations
- File conversion utilities

Description:

The HF2 Instrument contains a default configuration for all settings when delivered to the customer. This default configuration is loaded every time the HF2 Instrument is powered up. It provides an initial safe state of the instrument so that externally connected setups are not damaged nor impacted. This default configuration cannot be modified by the user.

However, ziControl provides the capability to load and save user specific configurations by means of this tab. As the configuration files are saved on the host computer, the user benefits from an unlimited number of possible configurations that can be stored.

The configuration files are in text format, thus they can be conveniently edited by the user. The syntax is self-explanatory and is the same as described for the text interface in Chapter Programming.

Saving demodulator samples to the hard disk can be done in text or binary format. Binary saving is faster and allows saving at higher sample rates, and minimizes CPU usage.

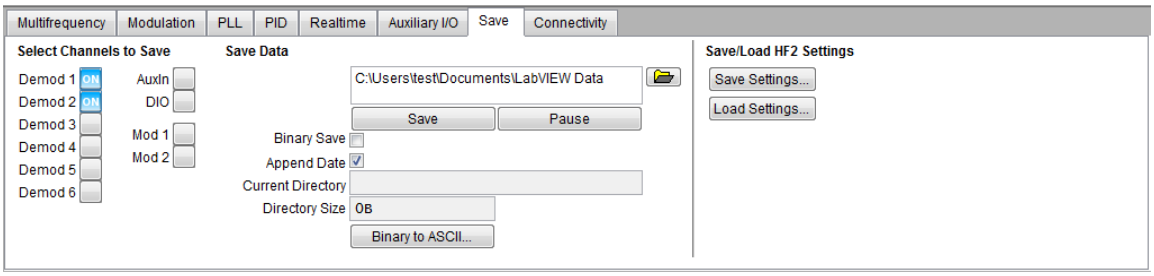


Figure 4.12. Save settings tab

Table 4.16. Save settings tab

Control/Tool	Options/Range	Description
Select channels to save	select Demodulators	set the demodulator outputs that are saved in CSV format to the specified directory on the host computer - one file is generated for each selected demodulator - refer to the next table to interpret the information regarding the data included in the files
	select Auxin	save data from the Auxiliary Inputs 1 and 2 to a separate file, at a maximum sampling rate of 400 kSa/s - note: the Auxiliary Inputs samples are also saved in the demodulator files at the readout rate of the demodulator - setting this switch generates large files
	select DIO	save data from the DIO port to a separate file, at a maximum sampling rate of 125 kSa/s - note: the DIO samples are also saved in the demodulator files at the readout rate of the demodulator - setting this switch may generate large files

Control/Tool	Options/Range	Description
	select Mod 1	save data from the AM/FM modulator 1 (requires related option to be activated)
	select Mod 2	save data from the AM/FM modulator 2 (requires related option to be activated)
Save directory	absolute path to the data save directory	set the trunk name for the save directory - note: you may browse to the target directory using the small icon at the top right
Binary save	OFF: ASCII save	-
	ON: binary save	
Append date to directory switch	OFF: no date	define whether a time-stamp is appended to the save directory name - this option allows to keep the same save path (trunk name) constant and to generate a unique directory name whenever a new save is started
	ON: append date	
Save button	OFF: stop the save	stop the save
	ON: start the save	start the save of demodulated samples to the directory defined in the save path field - pressing this button generates 1 file for each active demodulator
Pause button	OFF: no pause	pause the save in progress of demodulated data - this feature is useful to append several measurement sequences into the same files, thus reducing the number of generated files and directories - note: this button has no impact on the activity of the demodulators (measurement in progress)
	ON: save is paused	
Current directory	-	indicates the current save directory (useful in combination with the append date switch)
Directory size	-	size of the current save directory - this value is the sum of all saved files in a directory and is updated during a save in progress
Binary to ASCII button	-	indicates that the ziControl is not able to save all data to the specified directories, and that some data were discarded
Save Settings button	-	save current settings of the HF2
Load Settings button	-	load settings into the HF2

Table 4.17. Save field information

File	Data	Description
FreqX	Timestamp [s]	the time-stamp counter is a value in seconds that is initiated with power-up of the HF2 Instrument and continues to increment as long as it is powered - the expiration of the time-stamp is in the order of years
	Demodulator X output [V_{RMS}]	the X output of the related demodulator at the set output rate
	Demodulator Y output [V_{RMS}]	the Y output of the related demodulator at the set output rate

File	Data	Description
	Frequency [Hz]	the frequency of the demodulator
	DIO [decimal unsigned integer]	the value of digital I/O port
	Auxiliary Input 1 [V]	the voltage at the auxiliary input 1
	Auxiliary Input 2 [V]	the voltage at the auxiliary input 2

Note

As the time-stamp information is consistent between all demodulators with the same readout rate, the user can merge and compare data from the various files.

4.3.4. Connectivity Settings Tab

Features:

- HF2 Instrument connectivity on local network - ziServer connectivity
- Selection of active instrument (in case more than one instrument is connected to the selected ziServer)
- Overview of installed product options and activation of new options
- Software and hardware revision information

Description:

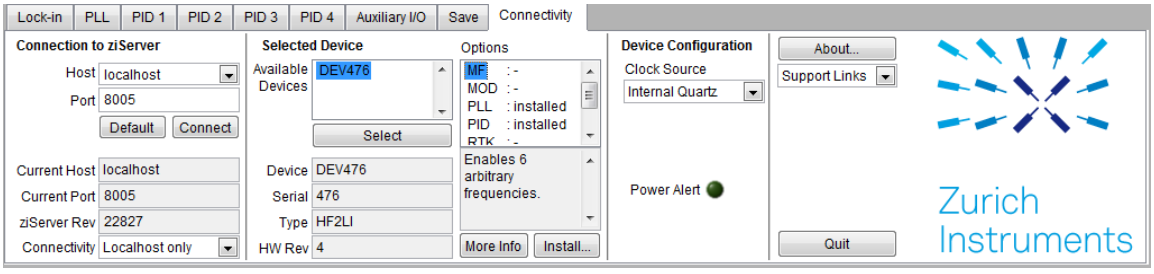


Figure 4.13. Connectivity settings tab

Table 4.18. Connectivity settings tab

Control/Tool	Options/Range	Description
Host	default: localhost, to be used when ziServer is running on the same host as ziControl	sets the host IP address where ziServer is running, and to which ziControl must connect in order to access the relevant HF2 Instrument
	range: any string recognized by your DNS	

Control/Tool	Options/Range	Description
Port	default: 8005	sets the port where ziServer is listening - the port number is given for a running server, and may be changed in the configuration files of ziServer
	range: 0 to 65535	
Default button	-	press this button to enter the default settings localhost:8005
Connect button	-	press this button to accept the entered host name and port
Current Host	-	information field regarding the connected host
Current Port	port number	information field regarding the connection port
ziServer Rev	revision number	information field regarding the revision of the connected server
Connectivity	Localhost only	the connected ziServer only accepts connectivity to the local host - remote clients are not able to access the local HF2 Instruments
	From everywhere	the connected ziServer accepts connectivity from any host on the LAN - all remote clients are able to access the local HF2 Instruments
Available Devices	select	this list shows all available HF2 Instruments on the selected server
Select button	-	press this button to select one of the available devices
Device	text	information field about the selected device
Serial	number	information field about the selected device
Type	text	HF2 Instrument type
HW Rev	text	hardware revision of the selected device
Installed options	MF	multi-frequency option is installed
	MOD	modulation option is installed
	PLL	phase-locked loop option is installed
	PID	PID option is installed
	RT	real-time option is installed
	UHS	ultra-high stability option is installed
More Info button	press	opens the product page on the Zurich Instruments website
Install options button	press	prompts a windows that permits to enter a feature code to install additional features
Clock Source selector	Internal Quartz	selects the internal oscillator as clock reference
	Clock In 10MHz	selects the external source for clock reference (external rubidium standard reference or atomic clock)
Power Alert	-	indicates that the supply voltage is too low (e.g. 100 V supply system) and therefore the instrument might provide incorrect measurements
About button	-	provides the information regarding the current version of ziControl

Control/Tool	Options/Range	Description
Support Links selector	User Manual	opens the installed version of the HF2 User Manual (this document)
	ZI Blogs	opens a browser window with access to the blog section of the Zurich Instruments website for a lot online content
	ZI Support	opens a browser window with access to the support section of the Zurich Instruments website for support information
	ZI Software Updates	opens a browser window with access to the software update information of the Zurich Instruments website
	Teamviewer Download Win	permits to download the teamviewer software for support in remote servicing (works also behind most firewalls, no need for software installation nor administrator rights)
Quit button	-	closes the ziControl application

4.3.5. Active Probes Settings Tab

The Active Probes settings tab adapts its content to the pre-amplifier connected to the HF2 Instrument. Whenever an active probe is connected with an RJ45 cable, this is automatically detected and a related screen appears on the ziControl tab. When the active probe is disconnected from the HF2, the panel on the tab disappears.

All active probes from Zurich Instruments are fully integrated inside ziControl.

HF2CA Current Amplifier

Features:

- Input impedance range from 10 V/A to 1 M V/A (R1, R2)
- Input mode differential or single-ended (Diff, Single)
- Input signal coupling mode (AC, DC)
- Output stage gain (G=1 or G=10)

Figure 4.14. Active probes HF2CA tab

Additional HF2CA specification can be found in the HF2CA Current Amplifier Datasheet .

HF2TA Current Amplifier

Features:

- Input offset +/- 10V
- Transimpedance gain from 100 V/A to 100M V/A (R1, R2)
- Input signal coupling mode (AC, DC)
- Addition gain (1, 10)
- Total gain display (R1*G, R2*G)
- Input Shield (GND, EXT Bias)
- Auxiliary output +/- 10V

Figure 4.15. Active probes HF2TA tab

Additional HF2TA specification can be found in the HF2TA Current Amplifier Datasheet .

4.4. Tools Tabs

4.4.1. Numerical Tool

Note

The number of available demodulators depends on the purchased instrument and activated options, and whether the features are activated in the related settings tabs. Polar, Cartesian, and Noise tools are available to all users. The AM/FM measurements only if the HF2LI-MOD option has been purchased and is activated in the related settings tabs

Features:

- Real-time demodulator output analysis
- Graphical and numerical range indicators
- Polar and Cartesian formats
- AM/FM modulation information (optional feature)
- Impedance measurement (optional feature)
- Support for arbitrary input unit function

Description:

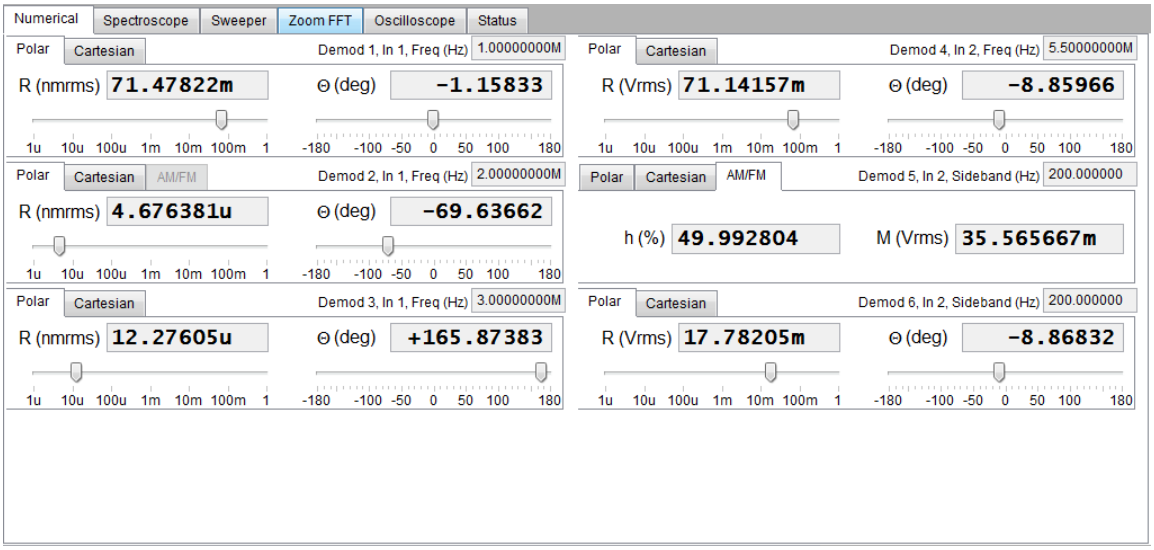


Figure 4.16. Numerical tool

Table 4.19. Polar and Cartesian tools

Control/Tool	Options/Range	Description
DEMOD1 polar	-	press this tab to see polar (R, Θ) sample representation for the demodulator - the amplitude R is indicated in V_{RMS} while the angle Θ in degrees
DEMOD1 Cartesian	-	press this tab to see Cartesian (X, Y) sample representation for the demodulator - the components X and Y are indicated in V_{RMS}

Control/Tool	Options/Range	Description
DEMOD1 AM/FM	-	press this tab to quickly measure the modulation index and the modulation
DEMOD1 frequency field	-	this field shows the current reference frequency of the demodulator
DEMOD1 X, Y, R, Θ fields	-	these fields indicate the value of the data samples as output by the demodulator filters - the update rate of these fields is equivalent to the readout rate setting - the outputs of the demodulators are in 64-bit resolution, thus much higher than represented on the screen - to profit from the full sample resolution, please save the samples to a file (save settings tab)
All other demodulators have the same functionality as DEMOD1		

Note

The following AM and FM tools descriptions are only valid for HF2LI models with HF2LI-MOD option activated.

The AM and FM tab is available in the tabs of demodulator 2 and demodulator 5 when the respective modulators are enabled in the modulations settings tab. The tab in demodulator 2 is for Mod 1, and the tab for demodulator 5 is for Mod 2.

Table 4.20. AM and FM tools

Control/Tool	Options/Range	Description
Modulation index (h %)	Amplitude modulation (AM)	this field shows the measured modulation index for amplitude modulation $h_{AM} = A_m$ / Ac (modulation amplitude divided by carrier amplitude)
	Frequency modulation (FM)	this field shows the measured modulation index for frequency modulation $h_{FM} = f_p / f_m$ (peak frequency divided by modulation frequency)
Modulation amplitude (V_{RMS})	Amplitude and frequency modulation	this field shows the cumulated amplitudes of the first sidebands (which are being demodulated)

4.4.2. Spectroscope Tool

Features:

- Real-time demodulator monitoring over time in graphical format
- Independent scale setting for each demodulator measured amplitude and phase angle, auto-scale
- Polar and Cartesian data format
- 2 cursors for data analysis

- Support for arbitrary input unit function

Description:

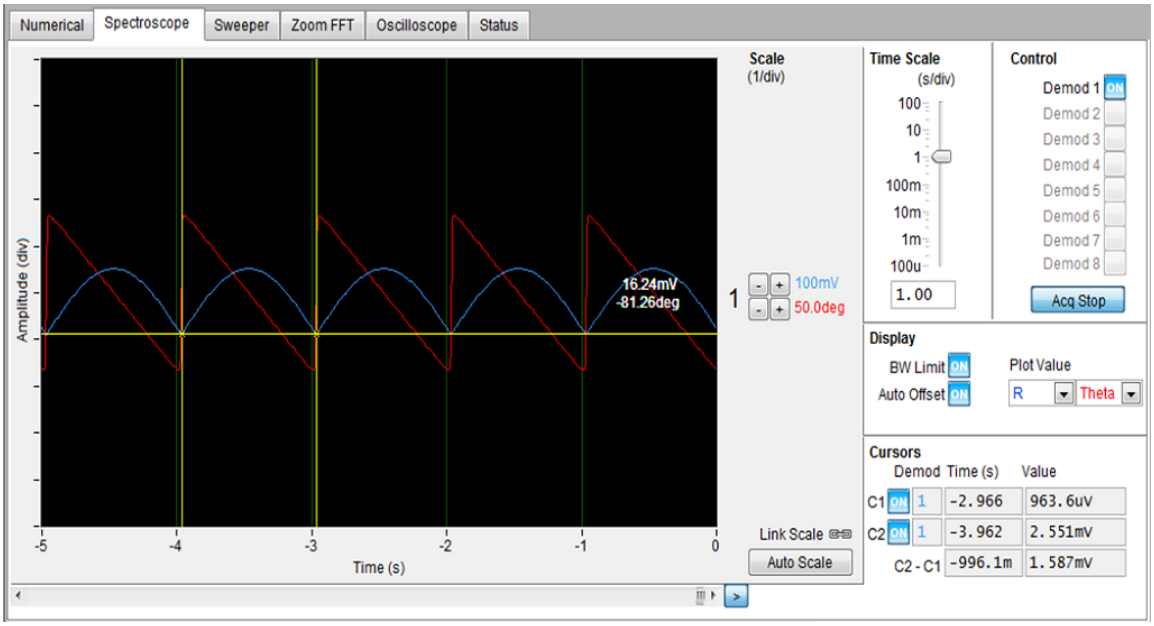


Figure 4.17. Spectroscope tool

Table 4.21. Spectroscope tool

Control/Tool	Options/Range	Description
Main spectroscope	-	shows a sample versus time plot - the last samples are plotted on the right, while older samples shift towards the left side - note that the horizontal axis always has 5 divisions, while the vertical axis has 10 divisions
Spectroscope time pan	[move slider]	move forward and backwards in time within the stored data
Spectroscope origin location (> button)	[press once]	forces the scope to stay right - the 0 seconds time always is fixed on the right hand side
Scale for X, Y, R, Θ display	V/DIV or degree/DIV	defines the voltage or degree per division that is displayed on vertical axis of the spectroscope screen for X/Y/R and Θ, respectively. The scale per division can be changed with no upper or lower bounds. Note: to change this value just press the +/- buttons or use the Auto Scale button
Link scale	[press once]	allows the manual vertical scaling to be done simultaneously for all demodulator plots
Auto scale button	[press once]	press once to adjust the vertical scale automatically
Scaling slider for time axis	100 µs/DIV to 100 s/DIV	defines the time per division that is displayed on the horizontal axis - in order to change the value move the slider, or directly enter a value in the box close to the slider, or press cursor up/down when the focus is inside that box

Control/Tool	Options/Range	Description
Demodulator controls	OFF: demodulator output not plotted	define whether the demodulator outputs are plotted in the spectroscope window - the number of plotted curves changes the location of the individual curves - only the demodulators that are turned on in the settings tabs can be selected here (the others are grayed out)
	ON: demodulator output plotted	
Acquisition stop button	OFF: spectroscope plotting ongoing	activates the plotting of the selected frequencies on the spectroscope window- this button stops and starts the plot of all frequencies at the same time
	ON: spectroscope plotting stopped	
BW limiter button	OFF: limiter off	inserts a low-pass filter which reduces the noise from higher-frequency by averaging
	ON: limiter on	
Auto offset button	OFF: keep offset	signals are plotted with constant offset
	ON: auto / no offset	signals are plotted with no offset
Plot Value	X, Y, R, Θ & Freq	select values to be plotted from two drop-down lists
Cursor controls	OFF: remove cursor	press to add a cursor - the cursor can be moved with the left mouse button
	ON: add cursor	
Cursor indicators	-	indicates current cursor values

4.4.3. Sweeper Tool

Features:

- Full-featured parametric sweep tool
- Full HF2 Instrument frequency range supported with many sweep modes: single, continuous (run/stop), forward, backward, bi-directional
- Overlap display of previous sweep results with persistent display
- Normalization of sweep with calibration control
- Auto bandwidth, averaging, and display normalization
- Fundamental and harmonic sweep support
- Parametric sweeper: Frequency, Phase, Time constant, Output amplitude, Offset (Aux Out)
- Support for arbitrary input unit function

Description:

The sweeper tool uses an user-selected reference signal as the excitation voltage and measures the frequency and phase response with an user-selected demodulator. This flexibility in demodulator selection permits to drive the sample-under-test with one frequency and to perform the measurement at either the fundamental or the harmonic of the driven frequency. The selected demodulators for reference and signal input may not be used for other measurements during the sweep.

In addition to the frequency sweep, the sweeper tool can also be used to obtain response from parametric sweeps such as phase, time constant, output amplitude as well as offset from the auxiliary outputs.

Figure 4.18. Sweeper tool

Table 4.22. Sweeper tool

Control/Tool	Options/Range	Description
Sweeper plot window	-	depicts the frequency response of the signal at signal input 1 or 2 - the measurement is made by placing a band-pass filter in discrete steps over the user-defined frequency range and simultaneously driving a sinusoidal signal at signal output 1 or 2 at the corresponding frequency
Sweeper Amp. label	User defined	default value is in Volt (V) - user defined unit can be enter in the Scale field under the Signal Inputs under the main lock-in control tab
Sweeper scale controls	Automatic	automatically fits the vertical and/or the horizontal scale to data (Automatic)
	Manual	scale is defined by the user
Show selection	Input Only	plots the measured input signal only
	Input / Ref	plots the input/ref quotient
	Impedance	plots various equivalent impedance model (ghosted out for HF2LI; only available for HF2IS)
	Ref Only	plots the reference signal only
Sweep range Start	1µ Hz to 50 MHz	defines the starting point of the freq. sweep
	-90 to 90 deg	defines the starting point of the phase sweep
	100 ps to 600 s	defines the starting point of the TC sweep
	-1 V to +1 V	defines the starting point of the amplitude sweep
	-10 V to +10 V	defines the starting point of the aux offset sweep
Sweep range Stop	1µ Hz to 100 MHz	defines the end point of the freq. sweep
	-90 deg to 90 deg	defines the end point of the phase sweep
	100 ps to 600 s	defines the end point of the TC sweep

Control/Tool	Options/Range	Description
	-1 V to +1 V	defines the end point of the amplitude sweep
	-10 V to +10 V	defines the end point of the aux offset sweep
Copy from Plot Range button	[press once]	after zooming into the plot, press once to adopt the start and stop frequencies from the plot
Sweep range Points	2 to 1'000'000	number of frequency steps (point) at which the sweeper performs its measurements - high values require long times to finish one sweep
Sweep range Direction	<->	bi-directional sweep (low-to-high and high-to-low values)
	->	sweep from low to high values
	<-	sweep from high to low values
Log Sweep button	OFF: linear sweep	samples are equidistant - plot scale is linear
	ON: logarithmic sweep	samples are logarithmic distributed - plot scale is logarithmic
Sweep control Run/ Stop button	OFF: stop	stop running continuous sweep
	ON: run	start continuous sweeping at the current frequency - after attaining the stop frequency, the sweeper restarts from the frequency depending on the sweep direction
Sweep control Single sweep button	OFF: stop	stop single sweep
	ON: start	start sweeping at the start frequency - after attaining the stop frequency, the sweeper stops
Sweep control Progress indicator	-	indicates the progress of the sweep
Sweep current value	-	indicates the current value during a sweep - the label changes depending on the type of parametric sweep being performed
Sweep value selection	Frequency	frequency sweep - oscillator selection 1 to 8 available
	Phase	phase sweep - output demodulator selection 1 to 6 available
	TC	time-constant sweep - output demodulator selection 1 to 6 available
	Amplitude	amplitude sweep - output demodulator selection 1 to 8 and signal output 1 to 2 available
	Aux Offset	offset sweep (Aux Out) - auxiliary output selection 1 to 4 available
	PID Setpoint	PID setpoint sweep - PID setpoint 1 to 4 selection available
Signal Input demod selection	1 to 6	sweeper uses the selected demodulator using its input choice, its oscillator choice, and its filter settings to perform the measurements of the sweep
Filter Settling time selection	5 TC eff	defines the sweeping time by filter settling time - for large bandwidths, the sweeping time is restricted by the software
	15 TC eff	
	30 TC eff	

Control/Tool	Options/Range	Description
Filter Averaging selection	1	defines the number of measurements taken at each point - the final value is calculated by averaging all measurements for the specific point
	2	
	4	
	8	
	16	
	32	
	64	
Filter Auto bandwidth control (active for frequency sweep)	OFF	manual bandwidth: the noise equivalent bandwidth (NEB) of the filter is defined by the settings of the chosen demodulator
	ON	auto bandwidth: the noise equivalent bandwidth is adjusted during the sweep to capture a larger part of the spectrum for logarithmic sweeps - this option has no effect for linear sweeps
Reference Mode selection	Off	reference mode is off
	Sig Out 1	sweeper uses signal output 1 as reference
	Sig Out 2	sweeper uses signal output 2 as reference
	Demod	sweeper uses one of the selected demodulators as reference
Reference Mode Demod selection	1 to 8	demodulator selection for reference mode (not available when Reference Mode is Off)
Display Coordinate system selection	Polar	selects the coordinate systems of the plot
	Polar (Log)	
	Cartesian	
	Nyquist	
Display Mapping selection	Linear	the Y scale of the plot is displayed with a linear unit
	dB	the Y scale of the plot is displayed with a logarithmic unit, being the ratio of the physical quantity (power and intensity) relative to a 1 V_{RMS} / V_{PK} reference level
	dBm	the Y scale of the plot is displayed with the logarithmic unit, being the power ratio referenced to 1 mW (dissipation resistor of 50 Ohm)
Display Result Unit selection (available for Show: Input Only and Show: Ref Only)	V_{RMS}	the Y scale of the plot reads the signal RMS voltage
	V_{PK}	the Y scale of the plot reads the signal peak voltage, $V_{PK} = \sqrt{2} * V_{RMS}$
	V_{RMS}^2	the Y scale of the plot reads the signal RMS power (squared RMS voltage)
	V_{PK}^2	the Y scale of the plot reads the signal peak power (squared peak voltage), $V_{PK}^2 = 2 * V_{RMS}^2$
	$V_{RMS}/\sqrt{Hz}$	the Y scale of the plot reads the spectral voltage density, independent of the frequency resolution

Control/Tool	Options/Range	Description
		- this setting is useful for noise measurements, $V_{RMS}/\sqrt{Hz} = V_{RMS} / \sqrt{(\text{frequency resolution})}$
	$V_{PK}/\sqrt{Hz}$	the Y scale of the plot reads the spectral peak density, independent of the frequency resolution - this setting is useful for noise measurements, $V_{PK}/\sqrt{Hz} = V_{PK} / \sqrt{(\text{frequency resolution})}$
	$V_{RMS}{}^2/Hz$	the Y scale of the plot reads the spectral RMS power density, independent of the frequency resolution - this setting is useful for noise measurements, $V_{RMS}{}^2/Hz = V_{RMS}$ / (frequency resolution)
	$V_{PK}{}^2/Hz$	the Y scale of the plot reads the spectral peak power density, independent of the frequency resolution - this setting is useful for noise measurements, $V_{PK}{}^2/Hz = V_{PK}$ / (frequency resolution)
Display Unwrap Phase switch	OFF / ON	enables the unwrapping of the phase in polar coordinate system
Equivalent Model selection (only visible when Show Impedance is selected in HF2IS)	|Z|, Phi (ohm)	display sweep result as absolute impedance and phase
	R, X (ohm)	display sweep result as real and imaginary impedance
	R//C	display sweep result as equivalent RC parallel circuit
	R, C	display sweep result as equivalent RC series circuit
	R, L	display sweep result as equivalent RL series circuit
	G, B	display sweep result as real and imaginary admittance
Persistent Display	Off	previous sweep tracking turned off
	Auto	allows to observe the differences between different plots
	Manual	allows to observe the differences between different plots and to memorize a plot of choice
Persistent Display Depth	1 to 100	number of traced sweeps (Note: each sweep will be displayed in a different color)
Persistent Display Persist	[press once]	memorizes current sweep (only active with History: Manual option)
Persistent Display Reset	[press once]	resets the history depth
Calibration Copy to Calib	[press once]	copies current sweep as the calibration reference
Calibration Control Amplitude button	[press once]	amplitude values will be calibrated
Calibration Control Phase button	[press once]	phase values will be calibrated

Control/Tool	Options/Range	Description
Calibration Save as... button	[press once]	saves the calibration setup to a file (XML format)
Calibration Load button	[press once]	loads the calibration setup from a file
Calibration selection	[drop-down menu]	allows to fast access the latest saved calibration setups
Q Factor	-	displays the estimated resonance Q from the frequency sweep
Save as... button	[press once]	generates a directory with the depicted sweeper plot as CSV and PNG files - a Readme.txt file including the description of the columns is saved along with the data - the user has the opportunity to define a directory name where the data is stored

Table 4.23. Sweeper tool plot scaling and cursor options

Control/Tool	Options/Range	Description
Plot X and Y scaling	Automatic	the X scale of the plot is calculated based on the FFT span, and Y scale of the plot is continuously adapted to the current demodulated samples
	Manual	the X scale and Y scale of the plot can be manually set - with manual setting, the cursors and zoom palette become available - the manual setting of the Y scaling can also be used to avoid the continuous changes in the scale limits
Plot cursor palette	[press once]	control permits to grab and move cursor point to the selected position
Plot zoom palette	[press once]	control permits to select one of 6 zoom modes including zoom in, zoom out, zoom X only, zoom Y only, and others
Cursor point control	2 to 1'000'000	defines the sample which the cursor is tracking
Cursor indicators	-	indicates current cursor values for X and Y scales
Cursor 2-1	-	indicates the difference between cursor 1 and 2

4.4.4. Zoom FFT Tool

Features:

- Fast, high-resolution FFT spectrum analyzer of demodulated data
- Variable center frequency, frequency resolution and frequency span
- Auto bandwidth, auto span (sampling rate)
- Choice of 4 different FFT window functions
- Continuous and block wise acquisition with different types of averaging

- Detailed noise power analysis
- Support for arbitrary input unit function

Description:

The Zoom FFT tool allows real-time FFT spectrum analysis on the complex samples (X+jY) output on each of the 6 demodulators. The Span of the calculated spectrum is equal to the Sampling rate (Hz) of the input Demodulator with a maximum value of 461 kHz. The resolution of the FFT spectra is determined by the duration recorded for each block of data, which is given by the ratio of the Span (Hz) and the number of Samples (#) recorded. With a maximum number of Samples (#) per block of 32.768 this leads for the largest span chosen still to a resolution of 14 Hz. When the frequency span is reduced to 112 Hz spectral features as close as 3.4 mHz can be resolved provided that the experimental setup is sufficiently stable over the recording time. Prior to calculating the complex FFT of the demodulator samples, one of the four different available window functions is applied to either optimize on dynamic range or resolution. The recommended default setting is using the Hann window which in general is considered to have a good trade off between both.

Technically, the center frequency of the FFT spectrum is always at zero when referred to the demodulator. However, taking the demodulator's reference oscillator into account the spectrum displayed can also be referred to the input signal by shifting it by the reference frequency, in the Zoom FFT tab indicated as Center frequency (Hz) of the FFT spectrum. Sometimes it is useful to scale the frequency axis accordingly by activating the Absolute freq button. Different spectral components on the input signal can then be attributed more easily to experiment parameters.

A basic signal analysis is performed where the signal at the center of the spectrum is determined and indicated as red curve in contrast to the noise and other spectral components drawn in white. For the Signal part of the spectrum a Noise Power Analysis is performed with all relevant values displayed in the lower right corner.

It is important to stress that the output of the demodulators are subject to low pass filtering and hence the always present noise floor will be naturally suppressed by this filter's frequency dependent transfer function. This behavior is indicated by the blue curve and labeled Fitted Noise. It resembles the calculated filter transfer function where the amplitude offset is fit with respect to the measured data. A very helpful feature is the Filter compensation which works by dividing the measurement data simply by the filter transfer function. This leads to a flat spectrum and allows for direct comparison of different amplitudes at various frequencies over the whole frequency span. Great care needs to be taken when activating the Filter Compensation that the SNR at the wings of the displayed spectrum is still sufficient. When using the auto setting button for the input sample rate one can be sure to be on the safe side here. This automatic algorithm targets to set the input sample rate to achieve a defined aliasing rejection rate of about 40 dB. This resembles the difference in dB between the filter transfer function in the center of the spectrum as compared to the very edge and can be simply measured from the highest point of the blue curve to the lowest points on the right and left borders of the FFT span while the Filter compensation is disabled. The FFT Advisor windows displays comments when settings chosen that lead to measurement artifact and misinterpretation of the data shown.

The noise panel provides useful information regarding the noise in the acquired demodulated signal. The noise tool inside ziControl always applies to demodulated samples. Consequently the noise figures depend on the bandwidth and the readout rate that is defined in the settings tabs. Measurements with large bandwidth collect more noise than measurements with small bandwidth.

Noise: $N = 1/n * \text{Sigma} (x_{AVG} - x)$

SNR: $SNR = 10 \times \log (R_{RMS} / N_{RMS})$

Averaging can be enabled to reduce the fluctuations in the display, but do not impact the noise measurement accuracy.

The noise spectral densities are calculated using the noise equivalent power bandwidth (NEPBW). The NEPBW is the effective bandwidth considering the area below the transfer function of a low-pass filter in the complete frequency spectrum. The NEPBW is actually larger than the typical 3 dB signal bandwidth.

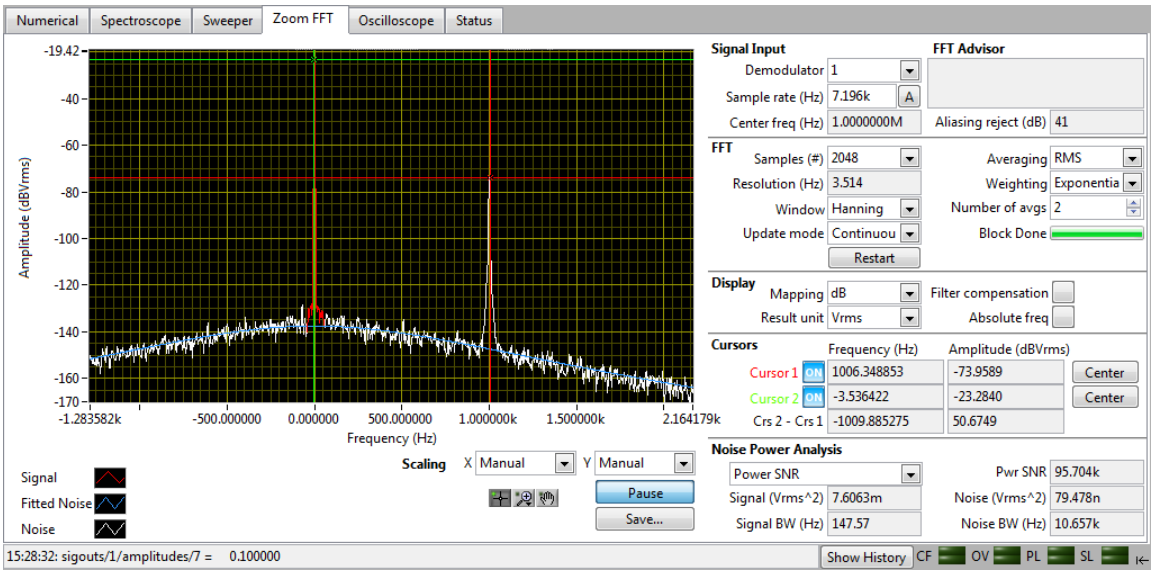


Figure 4.19. Zoom FFT tool

Table 4.24. Zoom FFT tool

Control/Tool	Options/Range	Description
FFT plot red line	Signal	the red plot line is the automatically determined signal component of the spectrum
FFT plot blue line	Fitted Noise	the blue plot line represents the filter transfer function of the selected demodulator fitted to the measured noise
FFT plot white line	Noise	the white plot line is the automatically determined noise component of the spectrum
Demodulator	1 to 6	determines the demodulator output that is taken for FFT spectrum analysis
Center frequency	-	indicates the center frequency of the selected demodulator - it is also the center frequency of the FFT span
FFT Advisor	-	provides information how to improve the quality of the spectrum plot and warnings
FFT Samples selection	256	defines the number of samples in one Block associated to the FFT span, and therefore also the frequency resolution - the number of selected samples together with the input sample rate determines the acquisition time for the FFT plot, T_{ACQ} = FFT_samples / signal_sample_rate
	512	
	1024	
	2048	
	4096	
	8192	
	16384	
	32786	

Control/Tool	Options/Range	Description
FFT Span (Hz)	0.22 Hz to 460 kHz	defines the demodulator output sample rate, equivalent to the FFT span
A = auto span	[press once]	performs an auto setting of the FFT span equivalent to the demodulator readout rate according to the settings of the demodulator filters (targets an aliasing rejection of 40 dB)
Resolution (Hz)	-	equivalent to the signal sample rate divided by the number of samples
FFT Window selection	Hann (default)	selection of different window functions for best signal representation
	Rectangle	
	Hamming	
	Blackman-Harris	
FFT Update Mode selection	Continuous	the FFT spectrum is updated approximatively every 100 ms
	Blocks	a new FFT is calculated after a new block of data is retrieved - when this mode is selected, a Block Done indicator will be displayed on the right-hand side
FFT Averaging selection	Off	disables the averaging for the FFT plot
	RMS averaging	enables RMS averaging for the FFT plot - the displayed plot is calculated by the average of the past plots (number defined by selector below)
	Peak hold	enables peak averaging for the FFT plot - the displayed plot represents the peak values detected since the start of the series
FFT Weighting	Linear	instructs the FFT plot to perform an average that equally weights the past plots - the acquisition of new FFT spectra stops after the defined number of averages
	Exponential	instructs the FFT plot to perform an average that weights recent plots more than older plots and keeps infinite history
FFT Number of averages	0 to 100.000	determines the number of blocks that are averaged over time to produce the shown plot (has effect on both linear and exponential averaging)
Block Done indicator	-	indicates the progress of samples recorded for one block (only available for FFT Update Mode set to Blocks)
Averages Done indicator	-	indicates the progress of averaging (only shown for Linear averaging) - the acquisition of new FFT spectra stops after the defined number of averages
Restart button	[press once]	press once to restart averaging
Result mapping selection	Linear	the Y scale of the plot is displayed with a linear unit
	dB	the Y scale of the plot is displayed with a logarithmic unit, being the ratio of the physical

Control/Tool	Options/Range	Description
		quantity (power and intensity) relative to a 1 V_{RMS} / V_{PK} reference level
	dBm	the Y scale of the plot is displayed with the logarithmic unit, being the power ratio referenced to 1 mW (dissipation resistor of 50 Ohm)
Result unit selection	$Unit_{RMS}$	the Y scale of the plot reads the signal RMS
	$Unit_{PK}$	the Y scale of the plot reads the signal peak, $Unit_{PK}$ = √2 * $Unit_{RMS}$
	$Unit_{RMS}^2$	the Y scale of the plot reads the signal RMS power (squared RMS Unit)
	$Unit_{PK}^2$	the Y scale of the plot reads the signal peak power (squared peak Unit), $Unit_{PK}^2$ = 2 * $Unit_{RMS}^2$
	$Unit_{RMS}$/√Hz	the Y scale of the plot reads the spectral Unit density, independent of the frequency resolution - this setting is useful for noise measurements, $Unit_{RMS}$/√Hz = $Unit_{RMS}$ / √(frequency resolution)
	$Unit_{PK}$/√Hz	the Y scale of the plot reads the spectral peak density, independent of the frequency resolution - this setting is useful for noise measurements, $Unit_{PK}$/√Hz = $Unit_{PK}$ / √(frequency resolution)
	$Unit_{RMS}^2$/Hz	the Y scale of the plot reads the spectral RMS power density, independent of the frequency resolution - this setting is useful for noise measurements, $Unit_{RMS}^2$/Hz = $Unit_{RMS}$ / (frequency resolution)
	$Unit_{PK}^2$/Hz	the Y scale of the plot reads the spectral peak power density, independent of the frequency resolution - this setting is useful for noise measurements, $Unit_{PK}^2$/Hz = $Unit_{PK}$ / (frequency resolution)
Filter compensation switch	OFF	the FFT plot does not compensate for the demodulator filter transfer function
	ON	the FFT plot compensates for the demodulator filter transfer function - this effect is small when the sample rate is in the same range as the demodulator bandwidth - the effect on the displayed plot is large when the signal sampling rate is much larger than the input signal rate
Absolute frequency switch	OFF	the X scale of the plot shows frequencies relative to the center of the FFT span which is labeled with 0 Hz (natural scaling)
	ON	the X scale of the plot is shifted by the demodulation frequency (Center freq) and refers to the frequencies applied to the Signal Input
Value selection	X, Y, R	selects the signal component that is taken for noise analysis
Value Mean ($Unit_{RMS}$)	-	displays the average of the number of the samples in one FFT plot

Control/Tool	Options/Range	Description
Value Noise ($Unit_{RMS}$)	-	displays the noise of the samples in one FFT plot
SNR	-	signal to noise ratio (SNR) calculated as the ratio between the value mean and the value noise
Value Noise Density ($Unit_{RMS}/\sqrt{Hz}$)	-	displays the noise of the samples in one FFT plot divided by the NEPBW of the demodulator
Pause button	OFF	continuous FFT plot
	ON	stopped FFT plot, the last calculated FFT plot is displayed and can be analyzed in detail (zoom and cursor palette)
Save ... button	[press once]	generates a directory with the depicted sweeper plot as CSV and PNG files - a Readme.txt file including the description of the columns is saved along with the data - the user has the opportunity to define a directory name where the data is stored

Table 4.25. Zoom FFT plot scaling and cursor options

Control/Tool	Options/Range	Description
Plot X and Y scaling	Automatic	the X scale of the plot is calculated based on the FFT span, and Y scale of the plot is continuously adapted to the current demodulated samples
	Manual	the X scale and Y scale of the plot can be manually set - with manual setting, the cursors and zoom palette become available - the manual setting of the Y scaling can also be used to avoid the continuous changes in the scale limits
Plot cursor palette	[press once]	control permits to grab and move cursor point to the selected position
Plot zoom palette	[press once]	control permits to select one of 6 zoom modes including zoom in, zoom out, zoom X only, zoom Y only, and others
Plot pan palette	[press once]	control permits to pan the plot with drag and drop - this is particularly useful for zoomed plots
Cursor 1 functions	ON	press to activate the cursor 1 - the plot must be paused in order to handle the cursor effectively
Cursor 2 functions	ON	press to activate the cursor 2 - the plot must be paused in order to handle the cursor effectively

4.4.5. Oscilloscope Tool

Features:

- Oscilloscope with 2048 samples memory for input and output signal monitoring
- 4 signal sources, up to 13 trigger sources, 16 different sampling rates, and 2 trigger methods

- Independent hold-off and trigger level settings
- Fast Fourier Transform (FFT), signal analysis, and sample histogram functions
- Support for arbitrary input unit function

Description:

The integrated oscilloscope with memory for 2048 samples provides a simple-to-use tool that allows to quickly check the signals at the inputs and outputs of the HF2 Instrument and to define the settings accordingly.

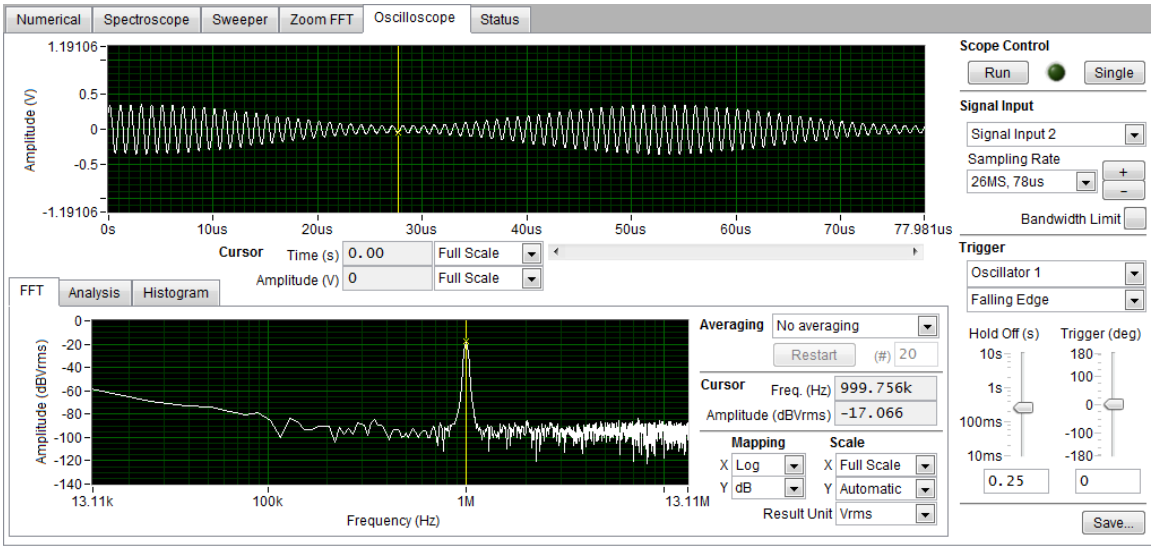


Figure 4.20. Oscilloscope tool with FFT tab

Table 4.26. Oscilloscope tool

Control/Tool	Options/Range	Description
Scope Control Run button	OFF / ON	activates the trigger of the oscilloscope for continuous oscilloscope waveforms
Trigger indicator	-	indicates that the oscilloscope has been triggered and that the display has been updated with a new waveform
Scope Control Single button	[press once]	activates the trigger of the oscilloscope for 1 single trigger event
Signal Input / oscilloscope source select	Signal Input 1	sets the oscilloscope to display the values sampled on Signal Input 1 after the HS-ADC
	Signal Input 2	sets the oscilloscope to display the values sampled on Signal Input 2 after the HS-ADC
	Signal Output 1	sets the oscilloscope to display the calculated values of Signal Output 1 before the HS-DAC - note: Add 1 does not impact this measured signal
	Signal Output 2	sets the oscilloscope to display the calculated values of Signal Output 2 before the HS-DAC - note: Add 2 does not impact this measured signal
Sampling Rate select	select	this control affects the sampling rate of the oscilloscope - changing the sampling rate impacts the frequencies that can be displayed without aliasing and the quality of the plot - thus this setting requires the user to consider

Control/Tool	Options/Range	Description
		the frequency components in his signal - this control also impacts the length in seconds of the displayed waveform (in Full Scale mode) - note: to change the value press the nearby increment/ decrement buttons, or use the pull-down select
	210 MS, 10 us	105 MHz maximum frequency, 10 us full scale plot
	105 MS, 20 us	52 MHz maximum frequency, 20 us full scale plot
	53 MS, 39 us	21 MHz maximum frequency, 39 us full scale plot
	26 MS, 78 us	13 MHz maximum frequency, 78 us full scale plot
	13 MS, 160 us	6.5 MHz maximum frequency, 160 us full scale plot
	6.6 MS, 310 us	3.3 MHz maximum frequency, 320 us full scale plot
	3.3 MS, 620 us	1.6 MHz maximum frequency, 620 us full scale plot
	1.6 MS, 1.2 ms	800 kHz maximum frequency, 1.2 ms full scale plot
	820 kS, 2.5 ms	410 kHz maximum frequency, 2.5 ms full scale plot
	410 kS, 5 ms	205 kHz maximum frequency, 5 ms full scale plot
	205 kS, 10 ms	100 kHz maximum frequency, 10 ms full scale plot
	103 kS, 20 ms	51 kHz maximum frequency, 20 ms full scale plot
	51 kS, 40 ms	25 kHz maximum frequency, 40 ms full scale plot
	26 kS, 80 ms	13 kHz maximum frequency, 80 ms full scale plot
	13 kS, 160 ms	6.5 kHz maximum frequency, 160 ms full scale plot
	6.4 kS, 320 ms	3.2 kHz maximum frequency, 320 ms full scale plot
Bandwidth Limit switch	OFF	deactivates averaging permitting to plot also higher frequencies - the plot window may be disturbed by aliasing effects (high frequency components that are down-mixed)
	ON	activates averaging reducing aliasing due to frequency components that are higher than the Nyquist bandwidth of the set sampling frequency - this switch reduces high-frequent noise components
Trigger source select	Continuous	a new waveform is acquired and displayed after the hold off time - the trigger phase is ignored
	Signal Input 1	a new waveform is acquired and displayed when Signal Input 1 matches the trigger condition
	Signal Input 2	a new waveform is acquired and displayed when Signal Input 2 matches the trigger condition
	Signal Output 1	a new waveform is acquired and displayed when Signal Output 1 matches the trigger condition

Control/Tool	Options/Range	Description
	Signal Output 2	a new waveform is acquired and displayed when Signal Output 2 matches the trigger condition
	Oscillator 1, 2, 3, 4, 5, 6, (7, 8)	a new waveform is acquired and displayed when the oscillator for frequency 1, 2, 3, 4, 5, 6, (7, 8) matches the trigger condition
	DIO 0/1	a new waveform is acquired and displayed when DIO 0/1 matches the trigger condition
Trigger slope select	Falling Edge	sets the trigger slope (has no meaning for continuous trigger)
	Rising Edge	
Hold Off slider	1 ms to 10 s	determines the time during which the trigger is deactivated after being triggered - the user can also enter the hold-off time manually, and increment or decrement the value using the keyboard arrow keys
Trigger phase slider	real (V or degree)	determines the analog value where the trigger fires and an oscilloscope waveform is acquired - the user can also enter the trigger level manually, and increment or decrement the value using the keyboard arrow keys
Scale X mode select	Full Scale	adjusts the X-scale so that all acquired oscilloscope points are plotted
	Manual Scale	activates the zoom palette for manual zooming into parts of the window
Scale Y mode select	Full Scale	adjusts the Y-scale to the range setting of the selected input signal
	Auto Scale	adjusts the Y-scale automatically to 150% of the maximum detected input level
	Manual Scale	activates the zoom palette for manual zooming into parts of the window
Cursor X position	-	displays the current position of the cursor in seconds
Cursor Y position	-	displays the current position of the cursor in the unit that is selected for the oscilloscope source
Save button	[press]	save the current waveform into a comma-separated values (CSV) file

Table 4.27. Oscilloscope/FFT tab

Control/Tool	Options/Range	Description
Averaging select	No averaging	FFT performed on the 2048 samples of the current waveform
	RMS averaging	RMS averaging reduces signal fluctuations but not the noise floor - the noise floor is not reduced because RMS averaging averages the energy, or power, of the signal - RMS averaging also causes averaged RMS quantities of single-channel measurements to have zero phase

Control/Tool	Options/Range	Description
	Peak hold	peak hold averaging retains the peak levels of the averaged quantities - peak hold averaging is performed at each frequency line separately, retaining peak levels from one FFT record to the next
Cursor position X and Y	-	displays the cursor position in Hz for the X coordinate and for the displayed unit on the Y coordinate
Mapping X select	Linear	sets the X axis to linear mode
	Manual	sets the X axis to logarithmic mode
Mapping Y select	Linear	sets the Y axis to linear mode
	dB	sets the Y axis to logarithmic mode displaying dB
	dBm	sets the Y axis to logarithmic mode displaying dBm
Scale X mode	Full Scale	the X axis of the FFT diagram is adjusted automatically to the scope sampling rate
	Manual	the X axis of the FFT diagram can be specified manually in the toolbox that appears - the X scale can also be specified by clicking directly on the values on the X-axis
Scale Y mode	Automatic	the Y scale of the FFT diagram is adapted automatically
	Manual	the Y scale of the FFT diagram can be specified manually in the toolbox that appears - the Y scale can also be specified by clicking directly on the values on the Y-axis
Result Unit	$Unit_{RMS}$	RMS of selected oscilloscope unit
	$Unit_{PK}$	peak of selected oscilloscope unit
	$Unit_{RMS}^2$	RMS power of selected oscilloscope unit
	$Unit_{PK}^2$	peak power of selected oscilloscope unit
	$Unit_{RMS}/\sqrt{Hz}$	RMS density of selected oscilloscope unit
	$Unit_{PK}/\sqrt{Hz}$	peak density of selected oscilloscope unit
	$Unit_{RMS}^2/Hz$	RMS power density of selected oscilloscope unit
	$Unit_{PK}^2/Hz$	peak power density of selected oscilloscope unit

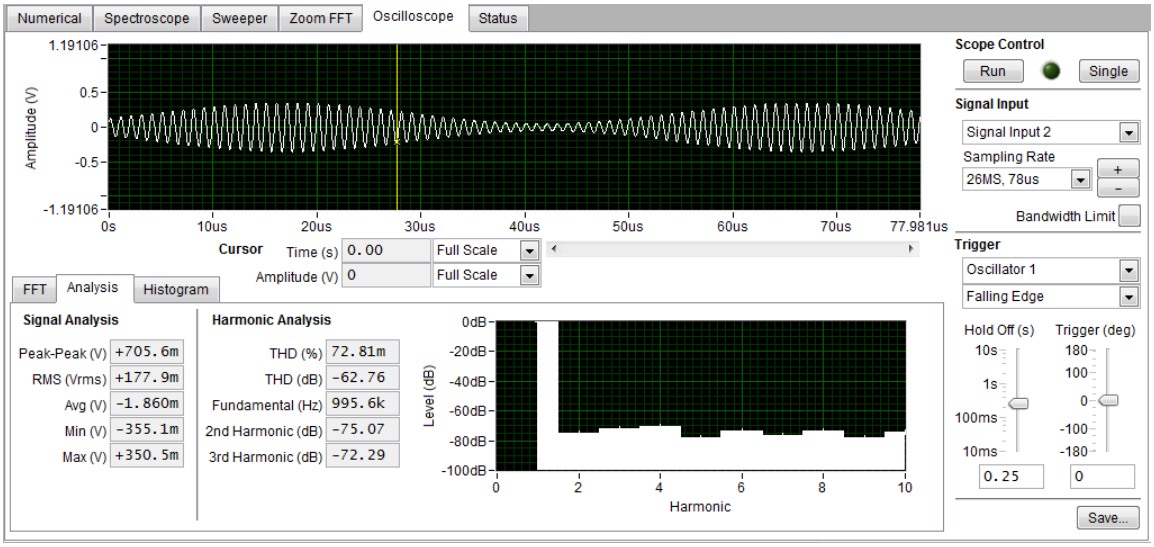


Figure 4.21. Oscilloscope tool with Analysis tab

Table 4.28. Oscilloscope/Analysis tab

Control/Tool	Options/Range	Description
Signal Peak-Peak value	-	maximum to minimum value of the samples in the displayed waveform
Signal RMS value	-	the root-mean-square of the samples in the displayed waveform
Signal Avg field	-	the average value of the samples in the displayed waveform
Signal Min field	-	the minimum value of the samples in the displayed waveform
Signal Max field	-	the maximum value of the samples in the displayed waveform
Total Harmonic Distortion (THD) field	-	the total harmonic distortion of the samples in the displayed waveform
Total Harmonic Distortion (THD) diagram	-	the total harmonic distortion diagram depicted with the harmonics in the X scale

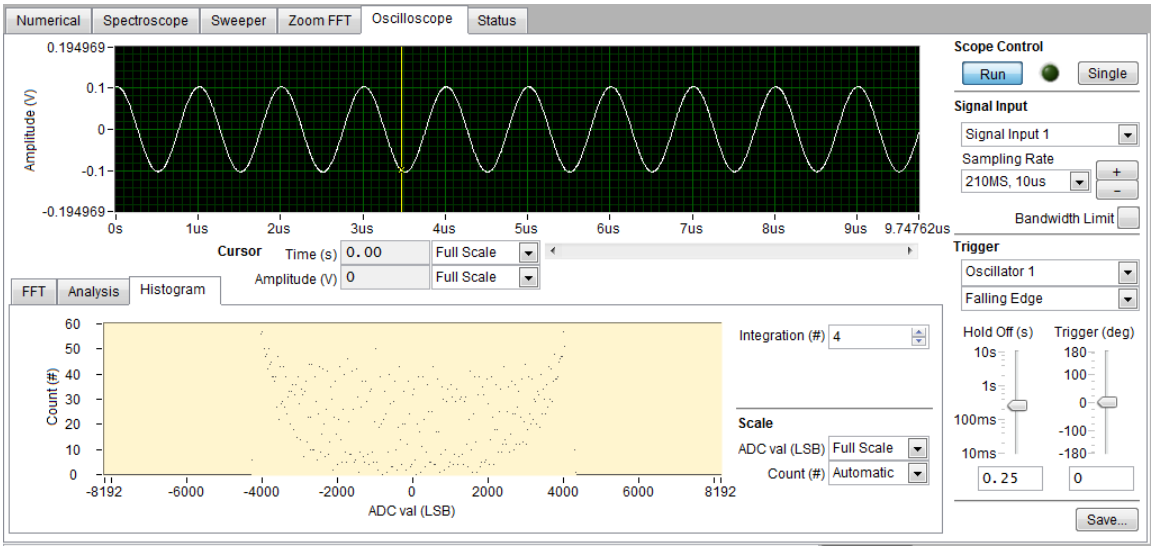


Figure 4.22. Oscilloscope tool with Histogram tab

Table 4.29. Oscilloscope/Histogram tab

Control/Tool	Options/Range	Description
Histogram display	-	shows the number of samples in the displayed waveforms with the value at the converter in the X scale
Integration select	1 to 256	defines the averaging on a number of waveforms, achieving a more stable histogram view
Scale X mode (ADC values)	Manual	makes the toolbox appear in order to set the X scale manually - select one scale label by clicking on it, and enter a new value - accept the new label by clicking somewhere in a neutral area of the GUI
	Full Scale	sets the X scale to full scale, equivalent to the resolution of the input ADC (from -8192 to 8192)
Scale Y mode (count values)	Automatic	sets the Y scale of the histogram display in automatic mode, so that the range is calculated depending on the actual waveform sample count
	Manual	makes the toolbox appear in order to set the Y scale manually - select one scale label by clicking on it, and enter a new value - accept the new label by clicking somewhere in a neutral area of the GUI

4.4.6. Status Tool and History Log

This section describes the features available in the status tab and in the history log section.

Features:

- First level instrument status information for problem solving (4 status indicators)

- Signal input level status, USB communication status
- Command history log

Description:

At the bottom-right area of the screen, 4 status indicators with 2 conditions each show notifications of the current status of the instrument. All status indicators indicate that a problem with the device is currently occurring or has occurred in the past. It is possible to reset the status indicators by pressing the return arrow symbol at the bottom-right. In case the status conditions existed in the past and are not true anymore, all indicators can be cleared at once. The clearing has not effect if the problem condition still exists: inspection of the current measurement conditions is required instead.

Warning

The data acquired when an error condition occurs may be faulty, incomplete, or inconsistent. Users should consider discarding the acquired data when a error condition has occurred during the measurement.

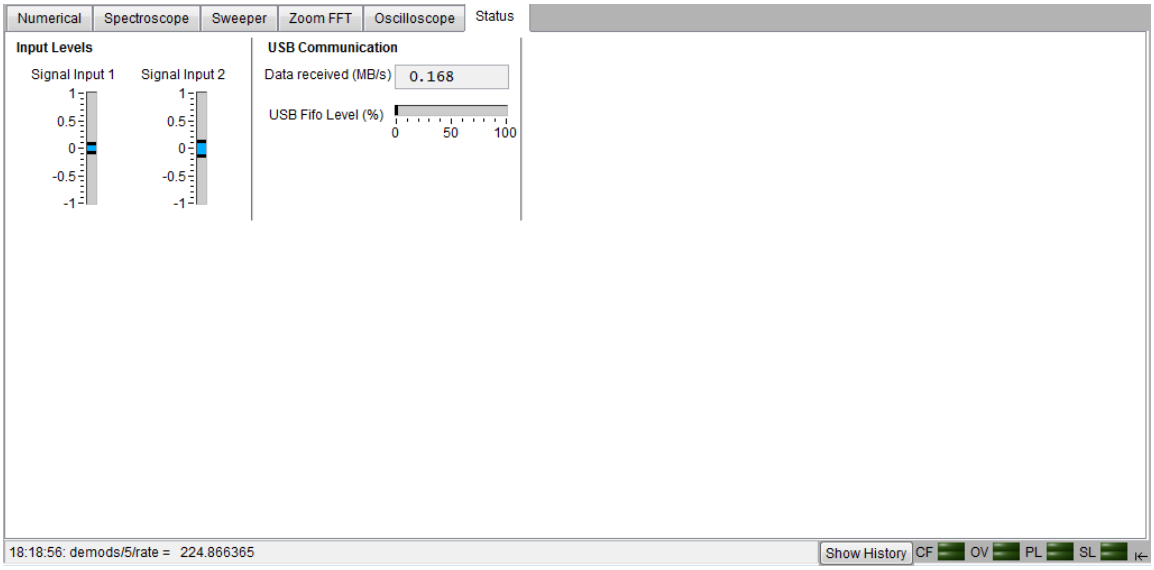


Figure 4.23. Status tool

The history log shows all commands that have been applied to the instrument since the start of the current ziControl session. There is a timestamp when the command has been performed and the command in the same syntax as for the text programming interface or for a LabVIEW program.

When clicking on the status line or on the Show History button, the complete history log opens and can be examined in detail. Copy and pasting of text parts is supported. Click again on the status line or the Hide History button to collapse the log.

The complete sequence of commands since the start of the ziControl session is also stored to a log file in the LabVIEW Data directory which is usually in this location `C:\Users\[USER]\Documents\LabVIEW Data` and the file name is called `com.zhinst.ziControlStatusLog.txt`.

Figure 4.24. History log

Table 4.30. Status tool

Control/Tool	Options/Range	Description
Input signal 1 level	-1 to +1	indicates in graphical format the current level at signal input 1 - the occurrence of an overshoot at signal input 1 lights the Input Over (OV) flag
Input signal 2 level	-1 to +1	indicates in graphical format the current level at signal input 2 - the occurrence of an overshoot at signal input 2 lights the Input Over (OV) flag
USB communication rate	[MB/s]	this level indicates the current communication throughput between the HF2 Instrument and the host computer - the rate is directly related to the readout rate of the demodulators - a level below 7 MB/s is suggested to avoid data loss
USB FIFO level	0% to 100%	the level indicates the current status of the USB FIFO buffer - a level above 100% during a short amount of time is equivalent to a package loss and the Package Loss (PL) flag is set - the FIFO level is dependent on many parameters, but in particular on the host computer and its operating system - higher performance computers are able to keep the FIFO level at a low level
CF (Clock fail)	Status	external clock is selected and does not work
	Occurred	clock fail has occurred in the past
OV (Input over voltage)	Status	the signals at Input 1 and/or Input 2 are overshooting the selected range and clipping of the samples occurs
	Occurred	input over voltage has occurred in the past
PL (Package loss)	Status	the USB communication is experiencing serious overflow and data packages are being discarded (samples and control) - this problem may be related to a high demodulator sample readout setting, however other reasons may also exist

Control/Tool	Options/Range	Description
	Occurred	package loss has occurred in the past
SL (Sample loss)	Status	the USB communication is experiencing serious demodulator sample losses - a sample loss may provoke or contribute to provoke a package loss - sample loss is less serious than package loss, however it may compromise the measurement integrity - a demodulator sample loss can be avoided by reducing the demodulator readout rate
	Occurred	sample loss has occurred in the past
Clear button	[press once]	clears all error flags

Chapter 5. Functional Description HF2IS

This section contains the detailed description of all panels of the graphical user interface (GUI) ziControl for the HF2IS product. This GUI is a LabVIEW based program that is delivered standard with all instruments.

On top of standard functionality like acquiring and saving data points this GUI provides a variety of measurement tools for the time and frequency domain. All of these features (and a few more) are also accessible by means of the programming interfaces described in Chapter Programming .

5.1. Graphical User Interface Overview

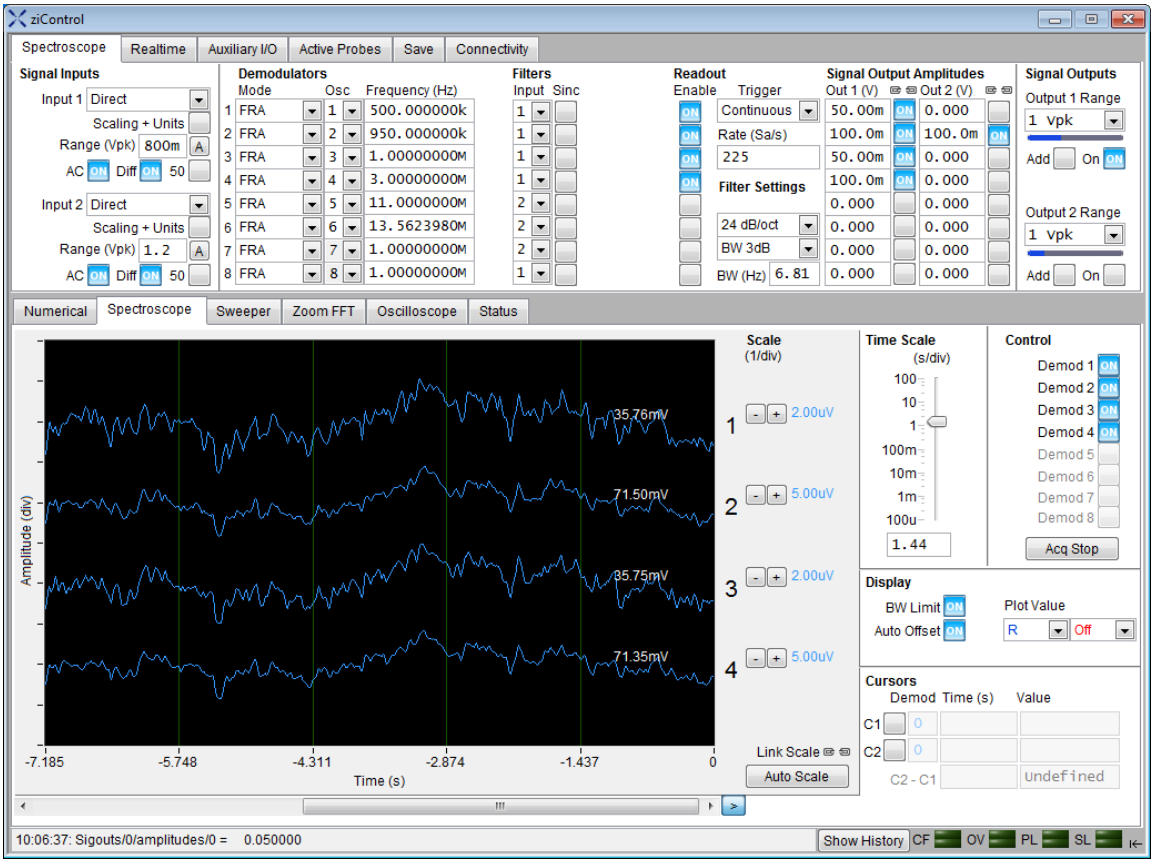


Figure 5.1. ziControl Overview

The GUI is divided into 2 sections as depicted in Figure 5.1 , each subdivided in a tab structure. The settings section on top is dedicated to display and control the main settings of the instrument, whereas the tools section provides additional measurement functionality. A status line on the lower end shows the command history which ziControl exchanged with the ziServer plus a couple of status flags on the right hand side.

The following tabs are available in the settings section:

- Spectroscope
- Real-time settings (optional feature)
- Auxiliary I/O settings
- Active probes settings (displayed only when external active probes are attached)
- Save settings
- Connectivity settings

These are the tabs in the tools section:

- Numerical tool
- Spectroscope tool
- Frequency response sweeper tool
- Zoom FFT tool

- Oscilloscope tool
- Status view

5.2. Settings Tabs

5.2.1. Spectroscope Tab

Note

Parts of this tab are ghosted when the HF2IS-MF multi-frequency option has not been purchased and is not activated.

Features:

- Control for 2 measurement units with 8 demodulators
- Control for 2 signal generators with up to 8 individual sinusoidal components
- Two and four terminal impedance measurement
- Continuous range setting for signal inputs and signal outputs
- Easy to use interface with many useful individual settings

Description:

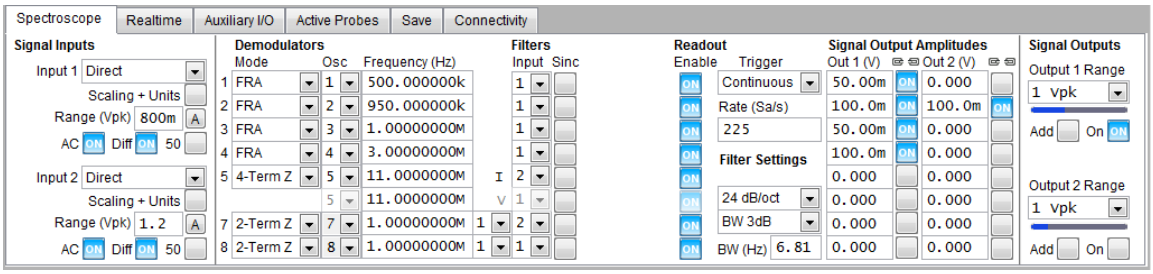


Figure 5.2. Spectroscope settings tab

Table 5.1. Spectroscope settings tab

Control/Tool	Options/Range	Description
Input 1/2 selection	Direct	defines the active probe that is connected to the input of the HF2 instrument - in case an HF2 Series pre-amplifier is connected then this selection will make the measurement account for external pre-amplification and current conversion - please make sure to define the correct pre-amplifier model, channel, and ZCtrl connectivity port (on the back panel of the instrument)
	HF2TA 0, Ch1	
	HF2TA 0, Ch2	
	HF2CA 0	
	HF2TA 1, Ch1	
	HF2TA 1, Ch2	
	HF2CA 1	
Scale + Unit enable	OFF / ON	enables arbitrary input scaling to define a scaling factor for all measurements regarding this input channel and also change the unit which is displayed throughout the tools and settings
Scale factor	numeric scaling factor	positive or negative real number to scale all measurement values attributed to that particular input; only appears when Scaling + Units is enabled

Control/Tool	Options/Range	Description
Unit acronym	three letter acronym	three letters can be set to define a unit, e.g. PSI; only appears when Scaling + Units is enabled
Range (unit)	1 mV to 2.1 V (or equivalent after unit + scale is applied)	sets the range of input 1/2 in fine increments within the available range - the value is the absolute range of the signal including a potential DC offset - note: the value inserted by the user may be approximated to the nearest value supported by the HF2; Scaling + Units is fully accounted for in this setting
A = Auto Range button	[press once]	press this button to automatically set the input range to two times the maximum amplitude of the input signal over a measured time of 100 ms
Signal input 1/2 AC switch	OFF: DC coupling	sets the input coupling for input 1/2 - an AC coupling inserts a high-pass filter at 1 kHz in the input path
	ON: AC coupling	
Signal input 1/2 differential switch	OFF: single-ended input	selects whether to connect to a single-ended signal (one cable) or to a differential signal (two cables)
	ON: differential input	
Signal input 1/2 50 Ω switch	OFF: 1 MΩ	sets the matching impedance for input 1/2
	ON: 50 Ω	
Demodulator 1-8 Mode selection	FRA	frequency response analyzer mode: the demodulator measures the RMS value of the selected unit
	2-Term Z	2 terminal impedance measurement mode: the demodulator measures impedance assuming a current is measured at the signal input and driven by the signal output voltage - note: this mode requires and external current amplifier to be connected between the DUT and the HF2 Instrument
	4-Term Z (demodulators 1,3,5 only)	4 terminal impedance measurement mode: the demodulator measures impedance assuming a current is measured at the signal input and driven by the signal output voltage - note: this mode requires and external current amplifier to be connected between the DUT and the HF2 Instrument - this mode also requires the use of two consecutive demodulators (1,2), (3,4), (5,6) - demodulators (7,8) cannot be used for 4-Term impedance measurement
Oscillator 1-6 selection	1 to 6	defines the active oscillator for each individual demodulator - note: the oscillator selection is be shaded in case 4-Term Z mode is selected for the demodulator above
	7 and 8	oscillators 7 and 8 are hardwired to demodulators 7 and 8
Oscillator 1-8 Frequency selection	0 to 100 MHz	sets the frequency for the corresponding generator

Control/Tool	Options/Range	Description
Demodulator 1-8 Signal Output selection	1: the demodulator is connected to Signal Output 1	sets the Signal Output that is taken for impedance measurement - note: this selection only appears if 2-Term Z mode is selected
	1: the demodulator is connected to Signal Output 2	
Demodulator 1-8 Signal Input selection	1: the demodulator is connected to Signal Input 1	sets the Signal Input that is connected to each of the demodulators
	2: the demodulator is connected to Signal Input 2	
Sinc Filter 1-8 switch	OFF: Sinc filter disabled for corresponding demodulator	the Sinc filter is an additional filtering stage that permits to remove the omega and 2 times omega components - by using a large filter roll-off value, the user can effectively reduce the signal component at the second and third harmonics - for applications where large roll-off is not possible, the Sinc filter achieves the same effect - note: there are limitations regarding the maximum frequency and the frequency resolution allowed with Sinc filtering (see Table 4.2)
	ON: Sinc filter enabled for corresponding demodulator	
Demodulator 1/8 activity switch	OFF: demodulator inactive	sets the activity for the corresponding demodulator - when enabled demodulated samples are sent to the host computer at the rate defined in the readout rate field
	ON: demodulators active	
Trigger selection	Continuous	automatic internal trigger with the sample rate defined in the next control field
	DIO 0	samples are sent to the host computer depending on DIO 0 triggering
	DIO 1	samples are sent to the host computer depending on DIO 1 triggering
	DIO 0 or 1	samples are sent to the host computer depending on DIO 0 and 1 triggering
	Undefined	a complex trigger has been programmed to the HF2 Instrument by means of the programming interfaces - this mode cannot be selected by the user of ziControl, but is the result of a complex trigger definition - note: this is a read-only position
Trigger Mode selection	Rising Edge	1 data sample is sent to the host computer for a rising edge on the trigger input
	Falling Edge	1 data sample is sent to the host computer for a falling edge on the trigger input
	Both Edges	1 data sample is sent to the host computer for a rising edge and falling edge on the trigger input
	Gate High	data samples are sent to the host computer as long as the defined DIO 0/1 input is high - it is possible to determine the number of

Control/Tool	Options/Range	Description
		demodulated samples that are sent to the host computer after a trigger by setting the readout rate and modulating the pulse length on the DIO 0/1
	Gate Low	data samples are sent to the host computer as long as the defined DIO 0/1 input is low - it is possible to determine the number of demodulated samples that are sent to the host computer after a trigger by setting the readout rate and modulating the pulse length on the DIO 0/1
	Undefined	a complex trigger has been programmed to the HF2 Instrument by means of the programming interfaces - this mode cannot be selected by the user of ziControl, but is the result of a complex trigger definition - note: this is a read-only position
Readout sample rate	0.22 Hz to 460 kHz	sets the filter readout rate, equivalent to the demodulator samples (measured data points) that are sent to host computer per second - it is also the rate of data received by ziControl and saved to the computer hard disk - this setting has no impact on the sample rate on the physical auxiliary outputs - note: the value inserted by the user may be approximated to the nearest value supported by the HF2 (see Table 9.4)
Filter order select	1st: 6 dB/oct	sets the filter order for all active demodulators
	2nd: 12 dB/oct	
	3rd: 18 dB/oct	
	4th: 24 dB/oct	
	5th: 30 dB/oct	
	6th: 36 dB/oct	
	7th: 42 dB/oct	
	8th: 48 dB/oct	
Filter Notation selection	BW 3 dB: 80 µHz (filter order = 8) to 200 kHz (filter order = 1)	defines the filter property indication mode either in bandwidth or time constant notation - either of these differ by a scaling factor depending on the filter order - please refer to Table 10.1 for exact relation between TC, BW, and filter order
	BW NEP: 90 µHz to 319 kHz	
	TC eff: 783 ns to 1900 s	
	TC per order: 783 ns to 580 s	
Filter setting	bandwidth or time constant	displays the filter setting according to the notation defined above - note: the value inserted by the user may be approximated to the nearest value supported by the Instrument

Control/Tool	Options/Range	Description
Signal Output 1/2 Amplitude 1-8	0 V to 10 V	sets the output voltage of the corresponding oscillator - the minimum setting is 0 V, the next level above zero depends on the Signal Output range - note: the value inserted by the user may be approximated to the nearest value supported by the Instrument
Signal Output 1/2 Amplitude Enable 1-8	OFF: oscillator output disabled	sets the output enable for the corresponding oscillator
	ON: oscillator output enabled	
Signal Output 1/2 range	±10 mV	sets the output range for output 1/2 - this select determines the maximum output peak to peak range - this setting will make sure that no peaks above the setting are generated at the output, independent from the amplitude settings (V) - the output signal is clipped if the amplitude is higher than the range - if the range is changed to a value smaller than V, then V is automatically reduced to the new range
	±100 mV	
	±1 V	
	±10 V	
Signal Output 1/2 Add switch	OFF: Add disabled	the Add input allows to add an external analog signal to the internally generated signal - when disabled, the signal at the Add input is ignored
	ON: Add enabled	
Signal Output 1/2 On switch	OFF: output disabled	sets the On switch for the related output - when the output is enabled, then the HF2 signal output is active and the blue LED on the front panel lights-up
	ON: output enabled	

Table 5.2. Filter order, TC, BW relation

Filter order	Bandwidth (BW)	BW at time constant TC=1 ms	BW at time constant TC=213 ms
1st: 6dB/oct	1.000/(2*π*TC)	159 Hz	0.75 Hz
2nd: 12 dB/oct	0.644/(2*π*TC)	102 Hz	0.48 Hz
3rd: 18 dB/oct	0.510/(2*π*TC)	81.2 Hz	0.38 Hz
4th: 24 dB/oct	0.435/(2*π*TC)	69.2 Hz	0.33 Hz
5th: 30dB/oct	0.386/(2*π*TC)	61.4 Hz	0.29 Hz
6th: 36 dB/oct	0.350/(2*π*TC)	55.7 Hz	0.26 Hz
7th: 42 dB/oct	0.323/(2*π*TC)	51.4 Hz	0.24 Hz
8th: 48 dB/oct	0.301/(2*π*TC)	47.9 Hz	0.22 Hz

5.3. Other Settings

5.3.1. Real-time Settings Tab

Note

The real-time tab appears only when the HF2LI-RT / HF2IS-RT option has been purchased and activated. Customers can purchase the RT option at any time, whether when ordering their instrument or after delivery. This option can be activated by the user or by Zurich Instruments via remote servicing.

Features:

- Push button loading of real-time programs via the graphical user interface.
- Easy access to the embedded microprocessor's user registers to configure program parameters or view program output at run-time. Optional naming of user registers for easy identification, saving and opening of register names for persistent configuration across sessions.
- Displays real-time debugging information in the message log field box and/or allows the user to save the message log to file.
- Graphical CPU load indicator.

Description:

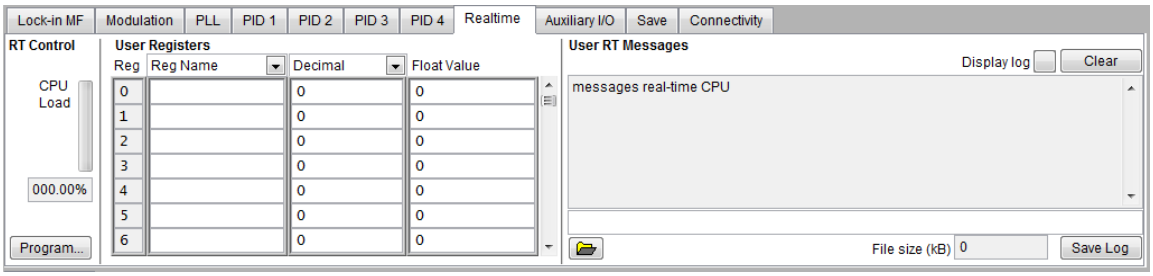


Figure 5.3. Real-time settings tab

Table 5.3. Real-time settings tab

Control/Tool	Options/Range	Description
CPU Load indicator	0% to 100%	indicates the momentary load of the embedded microprocessor running the real-time code
Program... push button	[press once]	selects a binary program to download to the HF2 Instrument; execution starts immediately thereafter
Reg Name pull-down menu	Reg Name	enters a name for the user register
	Save...	saves the user register names in an XML user register configuration file
	Open...	loads user register names saved in an XML user register configuration file
	Clear	clears entered user names. The values of the user register remain unaffected

Control/Tool	Options/Range	Description
Format pull-down menu	Decimal	selects the display format of the user register between decimal and hexadecimal
	Hexadecimal	
Float Value	IEEE754 32-bit floating-point number	view or edit a user register value as a 32-bit floating point number
Register fields 0-63	Register name & value	visualizes the processor register content - these fields are updated as the processor modifies the values - these fields are read/write: they permit to directly write values into the registers of the embedded processor. Each register can be optionally named for easy identification
User RT Messages	Read only	displays log messages printed via `ziRTKPrintf()` from the real-time program running on the embedded processor.
Display log toggle button	OFF	update the message log field disabled
	ON	update the message log field enabled
Clear push button	[press once]	clears the message log field
Log file name edit box	File name	specifies a destination file to save real-time log messages
Folder push button	[press once]	opens a dialog to specify a destination file to save real-time log messages
File size (kB)	Read only	displays the current size of the saved log file
Save log push button	[press once]	saves the real-time log messages in the specified file

5.3.2. Auxiliary I/O Settings Tab

Features:

- Control for auxiliary output connectors
- Monitor of auxiliary input connectors
- Monitor and control of digital I/O connectors

Description:

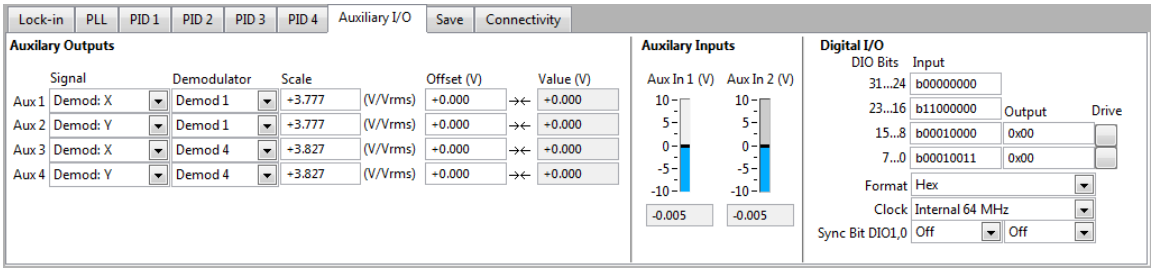


Figure 5.4. Auxiliary I/O settings tab

Table 5.4. Auxiliary I/O settings tab

Control/Tool	Options/Range	Description
Aux 1/2/3/4 output signal select	Manual (V)	set the signal that is provided on the Aux 1/2/3/4 connectors on the front panel of the instrument - the manual setting generates a DC value - the PLL:df selection is only available with the HF2LI-PLL option - the PID:Output selection is only available with the HF2LI-PID option
	X (V_{RMS})	
	Y (V_{RMS})	
	R (V_{RMS})	
	Theta (deg)	
	PLL1: dF (Hz)	
	PLL2: dF (Hz)	
	PID1: Output (PID output unit)	
	PID2: Output (PID output unit)	
	PID3: Output (PID output unit)	
	PID4: Output (PID output unit)	
Aux 1/2/3/4 output demodulator select	Demodulators	set the demodulator unit whose samples are being output to the related Aux output connector on the front panel of the instrument - note: this field is not available when the signal select is on manual or on PLL:dF or on PID:Output
Aux 1/2/3/4 output signal scale	-80G to 80G (V/V_{RMS}) for X , Y, R	set the scaling factor applied to the selection, consisting of a digital multiplication in real-time - the range of this field depends on the selected input range (see the following table) - note: this field is not available when the signal select is on manual
	-460k to 460k (V/deg) for Θ	
	-1.6 to 1.6 (V/Hz) for PLL:dF	
	-10^{37} to 10^{37} for PID:output	
Aux 1/2/3/4 output offset	-10 V to 10 V	set the DC offset value that is being added after the scaling of the selection
Aux 1/2/3/4 output value	[display]	instantaneous value at the corresponding auxiliary output connector
Aux In 1	-10 V to 10 V	indicates the current signal level on auxiliary input 1 (back panel)
Aux In 2	-10 V to 10 V	indicates the current signal level on auxiliary input 2 (back panel)
Digital I/O input bits 31...24	[input]	digital value in DIO input register bits 31 to 24
Digital I/O input bits 23...16	[input]	digital value in DIO input register bits 23 to 16
Digital I/O input bits 15...8	[input]	digital value in DIO input register bits 15 to 8
Digital I/O input bits 7...0	[input]	digital value in DIO input register bits 7 to 0

Control/Tool	Options/Range	Description
Digital I/O output bits 15...8	0x00 to 0xFF	digital value in DIO output register bits 15 to 8
Digital I/O output bits 7...0	0x00 to 0xFF	digital value in DIO output register bits 7 to 0
Digital I/O drive selector bits 15 to 8	OFF: no drive (input)	digital value at the DIO port bits 15 to 8 is not driven, high impedance, and the connectors can be used as inputs
	ON: drive (output)	digital value at the DIO port bits 15 to 8 is driven
Digital I/O drive selector bits 7 to 0	OFF: no drive (input)	digital value at the DIO port bits 7 to 0 is not driven, high impedance, and the connectors can be used as inputs
	ON: drive (output)	digital value at the DIO port bits 7 to 0 is driven
Digital I/O Hex/Bin	Hex	change between hexadecimal and binary representation
	Binary	
Digital I/O Clock	Internal 64 MHz	DIO input register is latched by internal 64 MHz clock
	Clk Pin 68	DIO input register is latched by signal on Pin 68 of the DIO port
Sync Bit DIO1,0	Off	DIO outputs 1 and 0 (BNC connectors) are not used for sync output and are free for other purposes
	Demod 1 to 8	reference signal of the selected demodulator is output on DIO 1 and 0 - note: there is a 166 ns delay between the sync and the front panel outputs (sync comes first) which leads to a relevant phase shift at high frequencies. To get an in phase sync, please make use of the WAVEFORM node setting and one of the HF front panel outputs

Table 5.5. Auxiliary output ranges

Input range setting	Scaling range for R (V/V_{RMS})	Scaling range for X and Y (V/V_{RMS})	Desired input full scale sensitivity (10 V full range output)
1 mV	80G	75G	1 nV to 1.5 V
10 mV	8G	7.5G	10 nV to 1.5 V
100 mV	760M	750M	100 nV to 1.5 V
1 V	83M	71M	1 µV to 1.5 V
2.1 V	42M	35M	2 µV to 1.5V

5.3.3. Save Settings Tab

Features:

- Save of demodulated samples to host computer disk drive
- Load and save of instrument configurations
- File conversion utilities

Description:

The HF2 Instrument contains a default configuration for all settings when delivered to the customer. This default configuration is loaded every time the HF2 Instrument is powered up. It provides an initial safe state of the instrument so that externally connected setups are not damaged nor impacted. This default configuration cannot be modified by the user.

However, ziControl provides the capability to load and save user specific configurations by means of this tab. As the configuration files are saved on the host computer, the user benefits from an unlimited number of possible configurations that can be stored.

The configuration files are in text format, thus they can be conveniently edited by the user. The syntax is self-explanatory and is the same as described for the text interface in Chapter Programming.

Saving demodulator samples to the hard disk can be done in text or binary format. Binary saving is faster and allows saving at higher sample rates, and minimizes CPU usage.

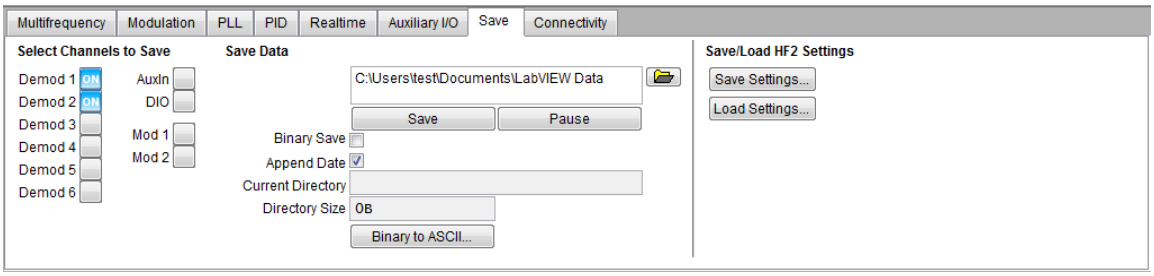


Figure 5.5. Save settings tab

Table 5.6. Save settings tab

Control/Tool	Options/Range	Description
Select channels to save	select Demodulators	set the demodulator outputs that are saved in CSV format to the specified directory on the host computer - one file is generated for each selected demodulator - refer to the next table to interpret the information regarding the data included in the files
	select Auxin	save data from the Auxiliary Inputs 1 and 2 to a separate file, at a maximum sampling rate of 400 kSa/s - note: the Auxiliary Inputs samples are also saved in the demodulator files at the readout rate of the demodulator - setting this switch generates large files
	select DIO	save data from the DIO port to a separate file, at a maximum sampling rate of 125 kSa/s - note: the DIO samples are also saved in the demodulator files at the readout rate of the demodulator - setting this switch may generate large files

Control/Tool	Options/Range	Description
	select Mod 1	save data from the AM/FM modulator 1 (requires related option to be activated)
	select Mod 2	save data from the AM/FM modulator 2 (requires related option to be activated)
Save directory	absolute path to the data save directory	set the trunk name for the save directory - note: you may browse to the target directory using the small icon at the top right
Binary save	OFF: ASCII save	-
	ON: binary save	
Append date to directory switch	OFF: no date	define whether a time-stamp is appended to the save directory name - this option allows to keep the same save path (trunk name) constant and to generate a unique directory name whenever a new save is started
	ON: append date	
Save button	OFF: stop the save	stop the save
	ON: start the save	start the save of demodulated samples to the directory defined in the save path field - pressing this button generates 1 file for each active demodulator
Pause button	OFF: no pause	pause the save in progress of demodulated data - this feature is useful to append several measurement sequences into the same files, thus reducing the number of generated files and directories - note: this button has no impact on the activity of the demodulators (measurement in progress)
	ON: save is paused	
Current directory	-	indicates the current save directory (useful in combination with the append date switch)
Directory size	-	size of the current save directory - this value is the sum of all saved files in a directory and is updated during a save in progress
Binary to ASCII button	-	indicates that the ziControl is not able to save all data to the specified directories, and that some data were discarded
Save Settings button	-	save current settings of the HF2
Load Settings button	-	load settings into the HF2

Table 5.7. Save field information

File	Data	Description
FreqX	Timestamp [s]	the time-stamp counter is a value in seconds that is initiated with power-up of the HF2 Instrument and continues to increment as long as it is powered - the expiration of the time-stamp is in the order of years
	Demodulator X output [V_{RMS}]	the X output of the related demodulator at the set output rate
	Demodulator Y output [V_{RMS}]	the Y output of the related demodulator at the set output rate

File	Data	Description
	Frequency [Hz]	the frequency of the demodulator
	DIO [decimal unsigned integer]	the value of digital I/O port
	Auxiliary Input 1 [V]	the voltage at the auxiliary input 1
	Auxiliary Input 2 [V]	the voltage at the auxiliary input 2

Note

As the time-stamp information is consistent between all demodulators with the same readout rate, the user can merge and compare data from the various files.

5.3.4. Connectivity Settings Tab

Features:

- HF2 Instrument connectivity on local network - ziServer connectivity
- Selection of active instrument (in case more than one instrument is connected to the selected ziServer)
- Overview of installed product options and activation of new options
- Software and hardware revision information

Description:

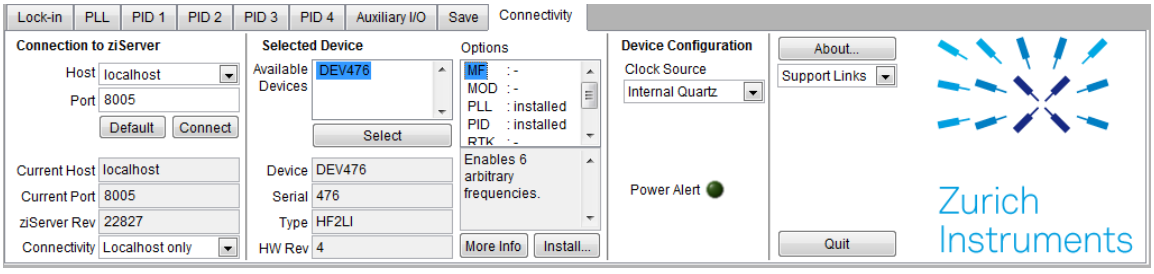


Figure 5.6. Connectivity settings tab

Table 5.8. Connectivity settings tab

Control/Tool	Options/Range	Description
Host	default: localhost, to be used when ziServer is running on the same host as ziControl	sets the host IP address where ziServer is running, and to which ziControl must connect in order to access the relevant HF2 Instrument
	range: any string recognized by your DNS	

Control/Tool	Options/Range	Description
Port	default: 8005	sets the port where ziServer is listening - the port number is given for a running server, and may be changed in the configuration files of ziServer
	range: 0 to 65535	
Default button	-	press this button to enter the default settings localhost:8005
Connect button	-	press this button to accept the entered host name and port
Current Host	-	information field regarding the connected host
Current Port	port number	information field regarding the connection port
ziServer Rev	revision number	information field regarding the revision of the connected server
Connectivity	Localhost only	the connected ziServer only accepts connectivity to the local host - remote clients are not able to access the local HF2 Instruments
	From everywhere	the connected ziServer accepts connectivity from any host on the LAN - all remote clients are able to access the local HF2 Instruments
Available Devices	select	this list shows all available HF2 Instruments on the selected server
Select button	-	press this button to select one of the available devices
Device	text	information field about the selected device
Serial	number	information field about the selected device
Type	text	HF2 Instrument type
HW Rev	text	hardware revision of the selected device
Installed options	MF	multi-frequency option is installed
	MOD	modulation option is installed
	PLL	phase-locked loop option is installed
	PID	PID option is installed
	RT	real-time option is installed
	UHS	ultra-high stability option is installed
More Info button	press	opens the product page on the Zurich Instruments website
Install options button	press	prompts a windows that permits to enter a feature code to install additional features
Clock Source selector	Internal Quartz	selects the internal oscillator as clock reference
	Clock In 10MHz	selects the external source for clock reference (external rubidium standard reference or atomic clock)
Power Alert	-	indicates that the supply voltage is too low (e.g. 100 V supply system) and therefore the instrument might provide incorrect measurements
About button	-	provides the information regarding the current version of ziControl

Control/Tool	Options/Range	Description
Support Links selector	User Manual	opens the installed version of the HF2 User Manual (this document)
	ZI Blogs	opens a browser window with access to the blog section of the Zurich Instruments website for a lot online content
	ZI Support	opens a browser window with access to the support section of the Zurich Instruments website for support information
	ZI Software Updates	opens a browser window with access to the software update information of the Zurich Instruments website
	Teamviewer Download Win	permits to download the teamviewer software for support in remote servicing (works also behind most firewalls, no need for software installation nor administrator rights)
Quit button	-	closes the ziControl application

5.3.5. Active Probes Settings Tab

The Active Probes settings tab adapts its content to the pre-amplifier connected to the HF2 Instrument. Whenever an active probe is connected with an RJ45 cable, this is automatically detected and a related screen appears on the ziControl tab. When the active probe is disconnected from the HF2, the panel on the tab disappears.

All active probes from Zurich Instruments are fully integrated inside ziControl.

HF2CA Current Amplifier

Features:

- Input impedance range from 10 V/A to 1 M V/A (R1, R2)
- Input mode differential or single-ended (Diff, Single)
- Input signal coupling mode (AC, DC)
- Output stage gain (G=1 or G=10)

Figure 5.7. Active probes HF2CA tab

Additional HF2CA specification can be found in the HF2CA Current Amplifier Datasheet .

HF2TA Current Amplifier

Features:

- Input offset +/- 10V
- Transimpedance gain from 100 V/A to 100M V/A (R1, R2)
- Input signal coupling mode (AC, DC)
- Addition gain (1, 10)
- Total gain display (R1*G, R2*G)
- Input Shield (GND, EXT Bias)
- Auxiliary output +/- 10V

Figure 5.8. Active probes HF2TA tab

Additional HF2TA specification can be found in the HF2TA Current Amplifier Datasheet .

5.4. Tools Tabs

5.4.1. Numerical Tool

Note

The number of available demodulators depends on the purchased instrument and activated options, and whether the features are activated in the related settings tabs. Polar, Cartesian, and Z tools are available to all users.

Features:

- Real-time demodulator output analysis
- Graphical and numerical range indicators
- Polar, Cartesian and Z formats
- Impedance measurement
- Support for arbitrary input unit function

Description:

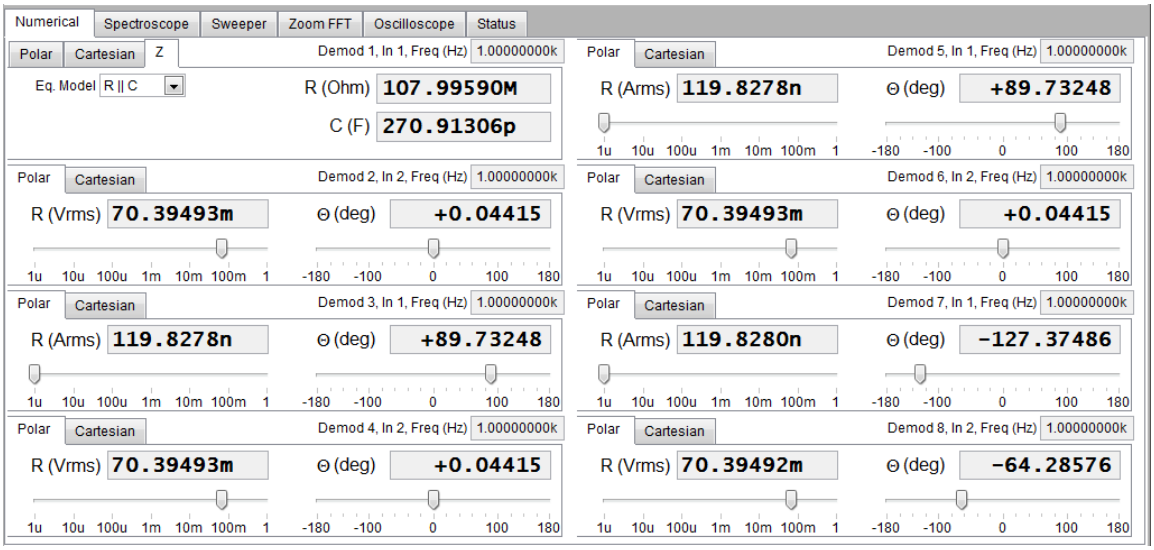


Figure 5.9. Numerical tool

Table 5.9. Polar and Cartesian tools

Control/Tool	Options/Range	Description
DEMOD1 polar	-	press this tab to see polar (R, Θ) sample representation for the demodulator - the amplitude R is indicated in V_{RMS} while the angle Θ in degrees
DEMOD1 Cartesian	-	press this tab to see Cartesian (X, Y) sample representation for the demodulator - the components X and Y are indicated in V_{RMS}

Control/Tool	Options/Range	Description
DEMOD1 Z	-	press this tab to display impedance sample representation. The available representations are: ▬ |Z|, Phi: impedance absolute value and phase ▬ ReZ, Imz: real and imaginary part of impedance ▬ R//C: resistor in parallel with capacitor ▬ R, C: resistor in series with capacitor ▬ D, C; dissipation factor and capacitance ▬ R, L: resistor in series with inductor ▬ G, B: conductance and susceptance (i.e. inverse of Z)
DEMOD1 frequency field	-	this field shows the current reference frequency of the demodulator
DEMOD1 X, Y, R, Θ, Z fields	-	these fields indicate the value of the data samples as output by the demodulator filters - the update rate of these fields is equivalent to the readout rate setting - the outputs of the demodulators are in 64-bit resolution, thus much higher than represented on the screen - to profit from the full sample resolution, please save the samples to a file (save settings tab)
All other demodulators have the same functionality as DEMOD1		

5.4.2. Spectroscope Tool

Features:

- Real-time demodulator monitoring over time in graphical format
- Independent scale setting for each demodulator measured amplitude and phase angle, auto-scale
- Polar and Cartesian data format
- 2 cursors for data analysis
- Support for arbitrary input unit function

Description:

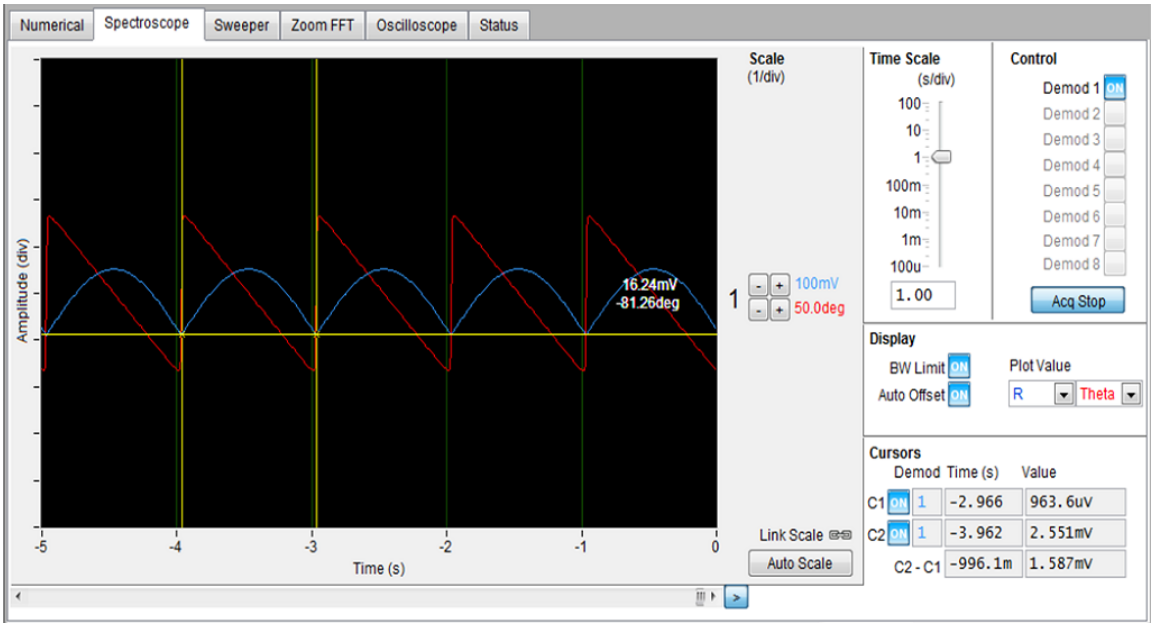


Figure 5.10. Spectroscope tool

Table 5.10. Spectroscope tool

Control/Tool	Options/Range	Description
Main spectroscope	-	shows a sample versus time plot - the last samples are plotted on the right, while older samples shift towards the left side - note that the horizontal axis always has 5 divisions, while the vertical axis has 10 divisions
Spectroscope time pan	[move slider]	move forward and backwards in time within the stored data
Spectroscope origin location (> button)	[press once]	forces the scope to stay right - the 0 seconds time always is fixed on the right hand side
Scale for X, Y, R, Θ display	V/DIV or degree/DIV	defines the voltage or degree per division that is displayed on vertical axis of the spectroscope screen for X/Y/R and Θ, respectively. The scale per division can be changed with no upper or lower bounds. Note: to change this value just press the +/- buttons or use the Auto Scale button
Link scale	[press once]	allows the manual vertical scaling to be done simultaneously for all demodulator plots
Auto scale button	[press once]	press once to adjust the vertical scale automatically
Scaling slider for time axis	100 µs/DIV to 100 s/DIV	defines the time per division that is displayed on the horizontal axis - in order to change the value move the slider, or directly enter a value in the box close to the slider, or press cursor up/down when the focus is inside that box
Demodulator controls	OFF: demodulator output not plotted	define whether the demodulator outputs are plotted in the spectroscope window - the number of plotted curves changes the location of the individual curves - only the demodulators that

Control/Tool	Options/Range	Description
	ON: demodulator output plotted	are turned on in the settings tabs can be selected here (the others are grayed out)
Acquisition stop button	OFF: spectroscope plotting ongoing	activates the plotting of the selected frequencies on the spectroscope window- this button stops and starts the plot of all frequencies at the same time
	ON: spectroscope plotting stopped	
BW limiter button	OFF: limiter off	inserts a low-pass filter which reduces the noise from higher-frequency by averaging
	ON: limiter on	
Auto offset button	OFF: keep offset	signals are plotted with constant offset
	ON: auto / no offset	signals are plotted with no offset
Plot Value	X, Y, R, Θ & Freq	select values to be plotted from two drop-down lists
Cursor controls	OFF: remove cursor	press to add a cursor - the cursor can be moved with the left mouse button
	ON: add cursor	
Cursor indicators	-	indicates current cursor values

5.4.3. Sweeper Tool

Features:

- Full-featured parametric sweep tool
- Full HF2 instrument frequency range supported with many sweep modes: single, continuous (run/stop), forward, backward, bi-directional
- Different impedance models display
- Overlap display of previous sweep results with persistent display
- Normalization of sweep with calibration control
- Auto bandwidth, averaging, and display normalization
- Parametric sweeper: Frequency, Phase, Time constant, Output amplitude, Offset (Aux Out)
- Support for arbitrary input unit function

Description:

The sweeper tool uses an user-selected reference signal as the excitation voltage and measures the frequency and phase response with an user-selected demodulator. This flexibility in demodulator selection permits to drive the sample-under-test with one frequency and to perform the measurement at either the fundamental or the harmonic of the driven frequency. The selected demodulators for reference and signal input may not be used for other measurements during the sweep.

In addition to the frequency sweep, the sweeper tool can also be used to obtain response from parametric sweeps such as phase, time constant, output amplitude as well as offset from the auxiliary outputs.

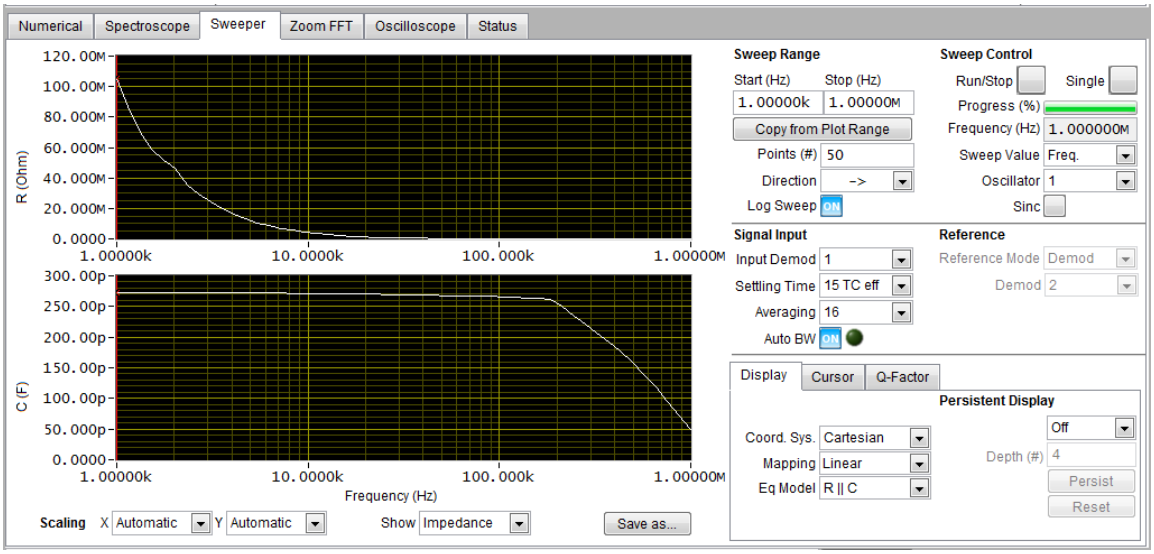


Figure 5.11. Sweeper tool

Table 5.11. Sweeper tool

Control/Tool	Options/Range	Description
Sweeper plot window	-	depicts the frequency response of the signal at signal input 1 or 2 - the measurement is made by placing a band-pass filter in discrete steps over the user-defined frequency range and simultaneously driving a sinusoidal signal at signal output 1 or 2 at the corresponding frequency
Sweeper Amp. label	User defined	default value is in Volt (V) - user defined unit can be enter in the Scale field under the Signal Inputs under the main lock-in control tab
Sweeper scale controls	Automatic	automatically fits the vertical and/or the horizontal scale to data (Automatic)
	Manual	scale is defined by the user
Show selection	Input Only	plots the measured input signal only
	Input / Ref	plots the input/ref quotient
	Impedance	plots various equivalent impedance model (only available for HF2IS)
	Ref Only	plots the reference signal only
Sweep range Start	1µ Hz to 50 MHz	defines the starting point of the freq. sweep
	-90 to 90 deg	defines the starting point of the phase sweep
	100 ps to 600 s	defines the starting point of the TC sweep
	-1 V to +1 V	defines the starting point of the amplitude sweep
	-10 V to +10 V	defines the starting point of the aux offset sweep
Sweep range Stop	1µ Hz to 100 MHz	defines the end point of the freq. sweep
	-90 deg to 90 deg	defines the end point of the phase sweep
	100 ps to 600 s	defines the end point of the TC sweep
	-1 V to +1 V	defines the end point of the amplitude sweep

Control/Tool	Options/Range	Description
	-10 V to +10 V	defines the end point of the aux offset sweep
Copy from Plot Range button	[press once]	after zooming into the plot, press once to adopt the start and stop frequencies from the plot
Sweep range Points	2 to 1'000'000	number of frequency steps (point) at which the sweeper performs its measurements - high values require long times to finish one sweep
Sweep range Direction	<->	bi-directional sweep (low-to-high and high-to-low values)
	->	sweep from low to high values
	<-	sweep from high to low values
Log Sweep button	OFF: linear sweep	samples are equidistant - plot scale is linear
	ON: logarithmic sweep	samples are logarithmic distributed - plot scale is logarithmic
Sweep control Run/ Stop button	OFF: stop	stop running continuous sweep
	ON: run	start continuous sweeping at the current frequency - after attaining the stop frequency, the sweeper restarts from the frequency depending on the sweep direction
Sweep control Single sweep button	OFF: stop	stop single sweep
	ON: start	start sweeping at the start frequency - after attaining the stop frequency, the sweeper stops
Sweep control Progress indicator	-	indicates the progress of the sweep
Sweep current value	-	indicates the current value during a sweep - the label changes depending on the type of parametric sweep being performed
Sweep value selection	Frequency	frequency sweep - oscillator selection 1 to 8 available
	Phase	phase sweep - output demodulator selection 1 to 6 available
	TC	time-constant sweep - output demodulator selection 1 to 6 available
	Amplitude	amplitude sweep - output demodulator selection 1 to 8 and signal output 1 to 2 available
	Aux Offset	offset sweep (Aux Out) - auxiliary output selection 1 to 4 available
	PID Setpoint	PID setpoint sweep - PID setpoint 1 to 4 selection available
Signal Input demod selection	1 to 6	sweeper uses the selected demodulator using its input choice, its oscillator choice, and its filter settings to perform the measurements of the sweep
Filter Settling time selection	5 TC eff	defines the sweeping time by filter settling time - for large bandwidths, the sweeping time is restricted by the software
	15 TC eff	
	50 TC eff	

Control/Tool	Options/Range	Description
Filter Averaging selection	1	defines the number of measurements taken at each point - the final value is calculated by averaging all measurements for the specific point
	2	
	4	
	8	
	16	
	32	
	64	
Filter Auto bandwidth control (active for frequency sweep)	OFF	manual bandwidth: the noise equivalent bandwidth (NEB) of the filter is defined by the settings of the chosen demodulator
	ON	auto bandwidth: the noise equivalent bandwidth is adjusted during the sweep to capture a larger part of the spectrum for logarithmic sweeps - this option has no effect for linear sweeps
Reference Mode selection	Off	reference mode is off
	Sig Out 1	sweeper uses signal output 1 as reference
	Sig Out 2	sweeper uses signal output 2 as reference
	Demod	sweeper uses one of the selected demodulators as reference
Reference Mode Demod selection	1 to 8	demodulator selection for reference mode (not available when Reference Mode is Off)
Display Coordinate system selection	Polar	selects the coordinate systems of the plot
	Polar (Log)	
	Cartesian	
	Nyquist	
Display Mapping selection	Linear	the Y scale of the plot is displayed with a linear unit
	dB	the Y scale of the plot is displayed with a logarithmic unit, being the ratio of the physical quantity (power and intensity) relative to a 1 V_{RMS} / V_{PK} reference level
	dBm	the Y scale of the plot is displayed with the logarithmic unit, being the power ratio referenced to 1 mW (dissipation resistor of 50 Ohm)
Display Result Unit selection (available for Show: Input Only and Show: Ref Only)	V_{RMS}	the Y scale of the plot reads the signal RMS voltage
	V_{PK}	the Y scale of the plot reads the signal peak voltage, $V_{PK} = \sqrt{2} * V_{RMS}$
	V_{RMS}^2	the Y scale of the plot reads the signal RMS power (squared RMS voltage)
	V_{PK}^2	the Y scale of the plot reads the signal peak power (squared peak voltage), $V_{PK}^2 = 2 * V_{RMS}^2$
	$V_{RMS}/\sqrt{Hz}$	the Y scale of the plot reads the spectral voltage density, independent of the frequency resolution

Control/Tool	Options/Range	Description
		- this setting is useful for noise measurements, $V_{RMS}/\sqrt{Hz} = V_{RMS} / \sqrt{(\text{frequency resolution})}$
	$V_{PK}/\sqrt{Hz}$	the Y scale of the plot reads the spectral peak density, independent of the frequency resolution - this setting is useful for noise measurements, $V_{PK}/\sqrt{Hz} = V_{PK} / \sqrt{(\text{frequency resolution})}$
	$V_{RMS}{}^2/Hz$	the Y scale of the plot reads the spectral RMS power density, independent of the frequency resolution - this setting is useful for noise measurements, $V_{RMS}{}^2/Hz = V_{RMS}$ / (frequency resolution)
	$V_{PK}{}^2/Hz$	the Y scale of the plot reads the spectral peak power density, independent of the frequency resolution - this setting is useful for noise measurements, $V_{PK}{}^2/Hz = V_{PK}$ / (frequency resolution)
Display Unwrap Phase switch	OFF / ON	enables the unwrapping of the phase in polar coordinate system
Equivalent Model selection (only visible when Show Impedance is selected in HF2IS)	|Z|, Phi (ohm)	display sweep result as absolute impedance and phase
	R, X (ohm)	display sweep result as real and imaginary impedance
	R//C	display sweep result as equivalent RC parallel circuit
	R, C	display sweep result as equivalent RC series circuit
	R, L	display sweep result as equivalent RL series circuit
	G, B	display sweep result as real and imaginary admittance
Persistent Display	Off	previous sweep tracking turned off
	Auto	allows to observe the differences between different plots
	Manual	allows to observe the differences between different plots and to memorize a plot of choice
Persistent Display Depth	1 to 100	number of traced sweeps (Note: each sweep will be displayed in a different color)
Persistent Display Persist	[press once]	memorizes current sweep (only active with History: Manual option)
Persistent Display Reset	[press once]	resets the history depth
Q Factor	-	displays the estimated resonance Q from the frequency sweep
Save as... button	[press once]	generates a directory with the depicted sweeper plot as CSV and PNG files - a Readme.txt file including the description of the columns is saved along with the data - the user has the opportunity

Control/Tool	Options/Range	Description
		to define a directory name where the data is stored

Table 5.12. Sweeper tool plot scaling and cursor options

Control/Tool	Options/Range	Description
Plot X and Y scaling	Automatic	the X scale of the plot is calculated based on the FFT span, and Y scale of the plot is continuously adapted to the current demodulated samples
	Manual	the X scale and Y scale of the plot can be manually set - with manual setting, the cursors and zoom palette become available - the manual setting of the Y scaling can also be used to avoid the continuous changes in the scale limits
Plot cursor palette	[press once]	control permits to grab and move cursor point to the selected position
Plot zoom palette	[press once]	control permits to select one of 6 zoom modes including zoom in, zoom out, zoom X only, zoom Y only, and others
Cursor point control	2 to 1'000'000	defines the sample which the cursor is tracking
Cursor indicators	-	indicates current cursor values for X and Y scales
Cursor 2-1	-	indicates the difference between cursor 1 and 2

5.4.4. Zoom FFT Tool

Features:

- Fast, high-resolution FFT spectrum analyzer of demodulated data
- Variable center frequency, frequency resolution and frequency span
- Auto bandwidth, auto span (sampling rate)
- Choice of 4 different FFT window functions
- Continuous and block wise acquisition with different types of averaging
- Detailed noise power analysis
- Support for arbitrary input unit function

Description:

The Zoom FFT tool allows real-time FFT spectrum analysis on the complex samples (X+jY) output on each of the 6 demodulators. The Span of the calculated spectrum is equal to the Sampling rate (Hz) of the input Demodulator with a maximum value of 461 kHz. The resolution of the FFT spectra is determined by the duration recorded for each block of data, which is given by the ratio of the Span (Hz) and the number of Samples (#) recorded. With a maximum number of Samples (#) per block of 32.768 this leads for the largest span chosen still to a resolution of 14

Hz. When the frequency span is reduced to 112 Hz spectral features as close as 3.4 mHz can be resolved provided that the experimental setup is sufficiently stable over the recording time. Prior to calculating the complex FFT of the demodulator samples, one of the four different available window functions is applied to either optimize on dynamic range or resolution. The recommended default setting is using the Hann window which in general is considered to have a good trade off between both.

Technically, the center frequency of the FFT spectrum is always at zero when referred to the demodulator. However, taking the demodulator's reference oscillator into account the spectrum displayed can also be referred to the input signal by shifting it by the reference frequency, in the Zoom FFT tab indicated as Center frequency (Hz) of the FFT spectrum. Sometimes it is useful to scale the frequency axis accordingly by activating the Absolute freq button. Different spectral components on the input signal can then be attributed more easily to experiment parameters.

A basic signal analysis is performed where the signal at the center of the spectrum is determined and indicated as red curve in contrast to the noise and other spectral components drawn in white. For the Signal part of the spectrum a Noise Power Analysis is performed with all relevant values displayed in the lower right corner.

It is important to stress that the output of the demodulators are subject to low pass filtering and hence the always present noise floor will be naturally suppressed by this filter's frequency dependent transfer function. This behavior is indicated by the blue curve and labeled Fitted Noise. It resembles the calculated filter transfer function where the amplitude offset is fit with respect to the measured data. A very helpful feature is the Filter compensation which works by dividing the measurement data simply by the filter transfer function. This leads to a flat spectrum and allows for direct comparison of different amplitudes at various frequencies over the whole frequency span. Great care needs to be taken when activating the Filter Compensation that the SNR at the wings of the displayed spectrum is still sufficient. When using the auto setting button for the input sample rate one can be sure to be on the safe side here. This automatic algorithm targets to set the input sample rate to achieve a defined aliasing rejection rate of about 40 dB. This resembles the difference in dB between the filter transfer function in the center of the spectrum as compared to the very edge and can be simply measured from the highest point of the blue curve to the lowest points on the right and left borders of the FFT span while the Filter compensation is disabled. The FFT Advisor windows displays comments when settings chosen that lead to measurement artifact and misinterpretation of the data shown.

The noise panel provides useful information regarding the noise in the acquired demodulated signal. The noise tool inside ziControl always applies to demodulated samples. Consequently the noise figures depend on the bandwidth and the readout rate that is defined in the settings tabs. Measurements with large bandwidth collect more noise than measurements with small bandwidth.

Noise: $N = 1/n * \text{Sigma} (x_{AVG} - x)$

SNR: $SNR = 10 \times \log (R_{RMS} / N_{RMS})$

Averaging can be enabled to reduce the fluctuations in the display, but do not impact the noise measurement accuracy.

The noise spectral densities are calculated using the noise equivalent power bandwidth (NEPBW). The NEPBW is the effective bandwidth considering the area below the transfer function of a low-pass filter in the complete frequency spectrum. The NEPBW is actually larger than the typical 3 dB signal bandwidth.

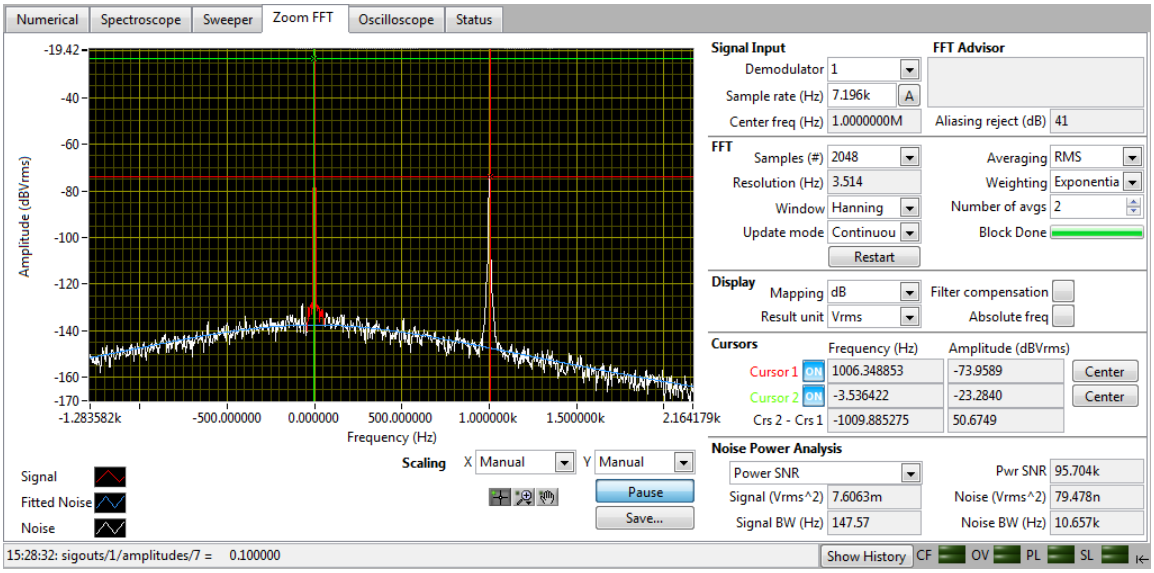


Figure 5.12. Zoom FFT tool

Table 5.13. Zoom FFT tool

Control/Tool	Options/Range	Description
FFT plot red line	Signal	the red plot line is the automatically determined signal component of the spectrum
FFT plot blue line	Fitted Noise	the blue plot line represents the filter transfer function of the selected demodulator fitted to the measured noise
FFT plot white line	Noise	the white plot line is the automatically determined noise component of the spectrum
Demodulator	1 to 6	determines the demodulator output that is taken for FFT spectrum analysis
Center frequency	-	indicates the center frequency of the selected demodulator - it is also the center frequency of the FFT span
FFT Advisor	-	provides information how to improve the quality of the spectrum plot and warnings
FFT Samples selection	256	defines the number of samples in one Block associated to the FFT span, and therefore also the frequency resolution - the number of selected samples together with the input sample rate determines the acquisition time for the FFT plot, T_{ACQ} = FFT_samples / signal_sample_rate
	512	
	1024	
	2048	
	4096	
	8192	
	16384	
	32786	
FFT Span (Hz)	0.22 Hz to 460 kHz	defines the demodulator output sample rate, equivalent to the FFT span
A = auto span	[press once]	performs an auto setting of the FFT span equivalent to the demodulator readout rate

Control/Tool	Options/Range	Description
		according to the settings of the demodulator filters (targets an aliasing rejection of 40 dB)
Resolution (Hz)	-	equivalent to the signal sample rate divided by the number of samples
FFT Window selection	Hann (default)	selection of different window functions for best signal representation
	Rectangle	
	Hamming	
	Blackman-Harris	
FFT Update Mode selection	Continuous	the FFT spectrum is updated approximatively every 100 ms
	Blocks	a new FFT is calculated after a new block of data is retrieved - when this mode is selected, a Block Done indicator will be displayed on the right-hand side
FFT Averaging selection	Off	disables the averaging for the FFT plot
	RMS averaging	enables RMS averaging for the FFT plot - the displayed plot is calculated by the average of the past plots (number defined by selector below)
	Peak hold	enables peak averaging for the FFT plot - the displayed plot represents the peak values detected since the start of the series
FFT Weighting	Linear	instructs the FFT plot to perform an average that equally weights the past plots - the acquisition of new FFT spectra stops after the defined number of averages
	Exponential	instructs the FFT plot to perform an average that weights recent plots more than older plots and keeps infinite history
FFT Number of averages	0 to 100.000	determines the number of blocks that are averaged over time to produce the shown plot (has effect on both linear and exponential averaging)
Block Done indicator	-	indicates the progress of samples recorded for one block (only available for FFT Update Mode set to Blocks)
Averages Done indicator	-	indicates the progress of averaging (only shown for Linear averaging) - the acquisition of new FFT spectra stops after the defined number of averages
Restart button	[press once]	press once to restart averaging
Result mapping selection	Linear	the Y scale of the plot is displayed with a linear unit
	dB	the Y scale of the plot is displayed with a logarithmic unit, being the ratio of the physical quantity (power and intensity) relative to a 1 V_{RMS} / V_{PK} reference level

Control/Tool	Options/Range	Description
	dBm	the Y scale of the plot is displayed with the logarithmic unit, being the power ratio referenced to 1 mW (dissipation resistor of 50 Ohm)
Result unit selection	$Unit_{RMS}$	the Y scale of the plot reads the signal RMS
	$Unit_{PK}$	the Y scale of the plot reads the signal peak, $Unit_{PK} = \sqrt{2} * Unit_{RMS}$
	$Unit_{RMS}^2$	the Y scale of the plot reads the signal RMS power (squared RMS Unit)
	$Unit_{PK}^2$	the Y scale of the plot reads the signal peak power (squared peak Unit), $Unit_{PK}^2 = 2 * Unit_{RMS}^2$
	$Unit_{RMS}/\sqrt{Hz}$	the Y scale of the plot reads the spectral Unit density, independent of the frequency resolution - this setting is useful for noise measurements, $Unit_{RMS}/\sqrt{Hz} = Unit_{RMS} / \sqrt{(frequency\ resolution)}$
	$Unit_{PK}/\sqrt{Hz}$	the Y scale of the plot reads the spectral peak density, independent of the frequency resolution - this setting is useful for noise measurements, $Unit_{PK}/\sqrt{Hz} = Unit_{PK} / \sqrt{(frequency\ resolution)}$
	$Unit_{RMS}^2/Hz$	the Y scale of the plot reads the spectral RMS power density, independent of the frequency resolution - this setting is useful for noise measurements, $Unit_{RMS}^2/Hz = Unit_{RMS} /$ (frequency resolution)
	$Unit_{PK}^2/Hz$	the Y scale of the plot reads the spectral peak power density, independent of the frequency resolution - this setting is useful for noise measurements, $Unit_{PK}^2/Hz = Unit_{PK} /$ (frequency resolution)
Filter compensation switch	OFF	the FFT plot does not compensate for the demodulator filter transfer function
	ON	the FFT plot compensates for the demodulator filter transfer function - this effect is small when the sample rate is in the same range as the demodulator bandwidth - the effect on the displayed plot is large when the signal sampling rate is much larger than the input signal rate
Absolute frequency switch	OFF	the X scale of the plot shows frequencies relative to the center of the FFT span which is labeled with 0 Hz (natural scaling)
	ON	the X scale of the plot is shifted by the demodulation frequency (Center freq) and refers to the frequencies applied to the Signal Input
Value selection	X, Y, R	selects the signal component that is taken for noise analysis
Value Mean ($Unit_{RMS}$)	-	displays the average of the number of the samples in one FFT plot
Value Noise ($Unit_{RMS}$)	-	displays the noise of the samples in one FFT plot

Control/Tool	Options/Range	Description
SNR	-	signal to noise ratio (SNR) calculated as the ratio between the value mean and the value noise
Value Noise Density ($Unit_{RMS}/\sqrt{Hz}$)	-	displays the noise of the samples in one FFT plot divided by the NEPBW of the demodulator
Pause button	OFF	continuous FFT plot
	ON	stopped FFT plot, the last calculated FFT plot is displayed and can be analyzed in detail (zoom and cursor palette)
Save ... button	[press once]	generates a directory with the depicted sweeper plot as CSV and PNG files - a Readme.txt file including the description of the columns is saved along with the data - the user has the opportunity to define a directory name where the data is stored

Table 5.14. Zoom FFT plot scaling and cursor options

Control/Tool	Options/Range	Description
Plot X and Y scaling	Automatic	the X scale of the plot is calculated based on the FFT span, and Y scale of the plot is continuously adapted to the current demodulated samples
	Manual	the X scale and Y scale of the plot can be manually set - with manual setting, the cursors and zoom palette become available - the manual setting of the Y scaling can also be used to avoid the continuous changes in the scale limits
Plot cursor palette	[press once]	control permits to grab and move cursor point to the selected position
Plot zoom palette	[press once]	control permits to select one of 6 zoom modes including zoom in, zoom out, zoom X only, zoom Y only, and others
Plot pan palette	[press once]	control permits to pan the plot with drag and drop - this is particularly useful for zoomed plots
Cursor 1 functions	ON	press to activate the cursor 1 - the plot must be paused in order to handle the cursor effectively
Cursor 2 functions	ON	press to activate the cursor 2 - the plot must be paused in order to handle the cursor effectively

5.4.5. Oscilloscope Tool

Features:

- Oscilloscope with 2048 samples memory for input and output signal monitoring
- 4 signal sources, up to 13 trigger sources, 16 different sampling rates, and 2 trigger methods
- Independent hold-off and trigger level settings
- Fast Fourier Transform (FFT), signal analysis, and sample histogram functions

- Support for arbitrary input unit function

Description:

The integrated oscilloscope with memory for 2048 samples provides a simple-to-use tool that allows to quickly check the signals at the inputs and outputs of the HF2 Instrument and to define the settings accordingly.

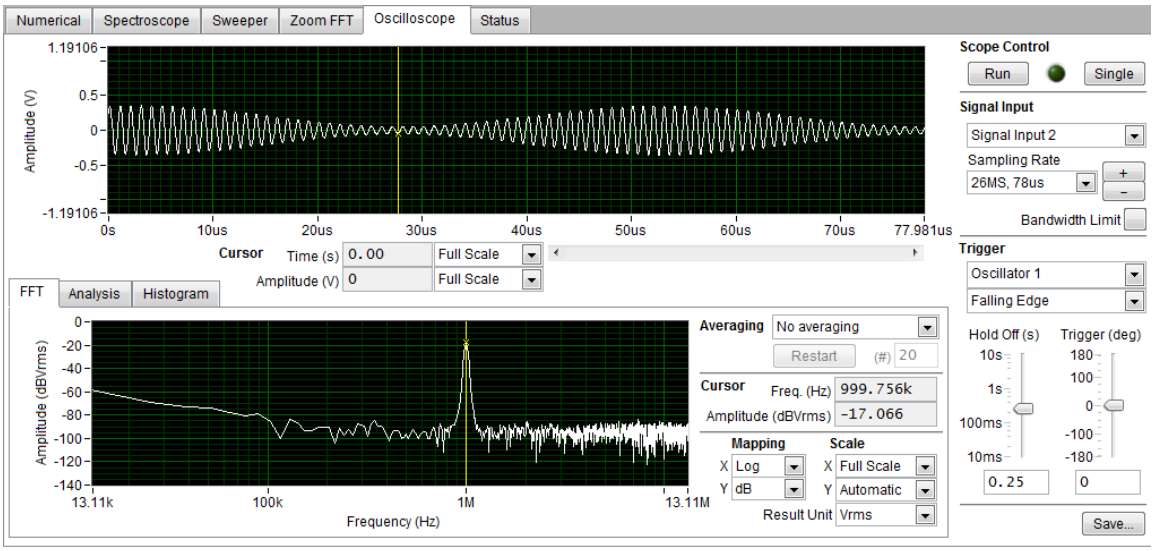


Figure 5.13. Oscilloscope tool with FFT tab

Table 5.15. Oscilloscope tool

Control/Tool	Options/Range	Description
Scope Control Run button	OFF / ON	activates the trigger of the oscilloscope for continuous oscilloscope waveforms
Trigger indicator	-	indicates that the oscilloscope has been triggered and that the display has been updated with a new waveform
Scope Control Single button	[press once]	activates the trigger of the oscilloscope for 1 single trigger event
Signal Input / oscilloscope source select	Signal Input 1	sets the oscilloscope to display the values sampled on Signal Input 1 after the HS-ADC
	Signal Input 2	sets the oscilloscope to display the values sampled on Signal Input 2 after the HS-ADC
	Signal Output 1	sets the oscilloscope to display the calculated values of Signal Output 1 before the HS-DAC - note: Add 1 does not impact this measured signal
	Signal Output 2	sets the oscilloscope to display the calculated values of Signal Output 2 before the HS-DAC - note: Add 2 does not impact this measured signal
Sampling Rate select	select	this control affects the sampling rate of the oscilloscope - changing the sampling rate impacts the frequencies that can be displayed without aliasing and the quality of the plot - thus this setting requires the user to consider the frequency components in his signal - this control also impacts the length in seconds of the

Control/Tool	Options/Range	Description
		displayed waveform (in Full Scale mode) - note: to change the value press the nearby increment/ decrement buttons, or use the pull-down select
	210 MS, 10 us	105 MHz maximum frequency, 10 us full scale plot
	105 MS, 20 us	52 MHz maximum frequency, 20 us full scale plot
	53 MS, 39 us	21 MHz maximum frequency, 39 us full scale plot
	26 MS, 78 us	13 MHz maximum frequency, 78 us full scale plot
	13 MS, 160 us	6.5 MHz maximum frequency, 160 us full scale plot
	6.6 MS, 310 us	3.3 MHz maximum frequency, 320 us full scale plot
	3.3 MS, 620 us	1.6 MHz maximum frequency, 620 us full scale plot
	1.6 MS, 1.2 ms	800 kHz maximum frequency, 1.2 ms full scale plot
	820 kS, 2.5 ms	410 kHz maximum frequency, 2.5 ms full scale plot
	410 kS, 5 ms	205 kHz maximum frequency, 5 ms full scale plot
	205 kS, 10 ms	100 kHz maximum frequency, 10 ms full scale plot
	103 kS, 20 ms	51 kHz maximum frequency, 20 ms full scale plot
	51 kS, 40 ms	25 kHz maximum frequency, 40 ms full scale plot
	26 kS, 80 ms	13 kHz maximum frequency, 80 ms full scale plot
	13 kS, 160 ms	6.5 kHz maximum frequency, 160 ms full scale plot
	6.4 kS, 320 ms	3.2 kHz maximum frequency, 320 ms full scale plot
Bandwidth Limit switch	OFF	deactivates averaging permitting to plot also higher frequencies - the plot window may be disturbed by aliasing effects (high frequency components that are down-mixed)
	ON	activates averaging reducing aliasing due to frequency components that are higher than the Nyquist bandwidth of the set sampling frequency - this switch reduces high-frequent noise components
Trigger source select	Continuous	a new waveform is acquired and displayed after the hold off time - the trigger phase is ignored
	Signal Input 1	a new waveform is acquired and displayed when Signal Input 1 matches the trigger condition
	Signal Input 2	a new waveform is acquired and displayed when Signal Input 2 matches the trigger condition
	Signal Output 1	a new waveform is acquired and displayed when Signal Output 1 matches the trigger condition

Control/Tool	Options/Range	Description
	Signal Output 2	a new waveform is acquired and displayed when Signal Output 2 matches the trigger condition
	Oscillator 1, 2, 3, 4, 5, 6, (7, 8)	a new waveform is acquired and displayed when the oscillator for frequency 1, 2, 3, 4, 5, 6, (7, 8) matches the trigger condition
	DIO 0/1	a new waveform is acquired and displayed when DIO 0/1 matches the trigger condition
Trigger slope select	Falling Edge	sets the trigger slope (has no meaning for continuous trigger)
	Rising Edge	
Hold Off slider	1 ms to 10 s	determines the time during which the trigger is deactivated after being triggered - the user can also enter the hold-off time manually, and increment or decrement the value using the keyboard arrow keys
Trigger phase slider	real (V or degree)	determines the analog value where the trigger fires and an oscilloscope waveform is acquired - the user can also enter the trigger level manually, and increment or decrement the value using the keyboard arrow keys
Scale X mode select	Full Scale	adjusts the X-scale so that all acquired oscilloscope points are plotted
	Manual Scale	activates the zoom palette for manual zooming into parts of the window
Scale Y mode select	Full Scale	adjusts the Y-scale to the range setting of the selected input signal
	Auto Scale	adjusts the Y-scale automatically to 150% of the maximum detected input level
	Manual Scale	activates the zoom palette for manual zooming into parts of the window
Cursor X position	-	displays the current position of the cursor in seconds
Cursor Y position	-	displays the current position of the cursor in the unit that is selected for the oscilloscope source
Save button	[press]	save the current waveform into a comma-separated values (CSV) file

Table 5.16. Oscilloscope/FFT tab

Control/Tool	Options/Range	Description
Averaging select	No averaging	FFT performed on the 2048 samples of the current waveform
	RMS averaging	RMS averaging reduces signal fluctuations but not the noise floor - the noise floor is not reduced because RMS averaging averages the energy, or power, of the signal - RMS averaging also causes averaged RMS quantities of single-channel measurements to have zero phase

Control/Tool	Options/Range	Description
	Peak hold	peak hold averaging retains the peak levels of the averaged quantities - peak hold averaging is performed at each frequency line separately, retaining peak levels from one FFT record to the next
Cursor position X and Y	-	displays the cursor position in Hz for the X coordinate and for the displayed unit on the Y coordinate
Mapping X select	Linear	sets the X axis to linear mode
	Manual	sets the X axis to logarithmic mode
Mapping Y select	Linear	sets the Y axis to linear mode
	dB	sets the Y axis to logarithmic mode displaying dB
	dBm	sets the Y axis to logarithmic mode displaying dBm
Scale X mode	Full Scale	the X axis of the FFT diagram is adjusted automatically to the scope sampling rate
	Manual	the X axis of the FFT diagram can be specified manually in the toolbox that appears - the X scale can also be specified by clicking directly on the values on the X-axis
Scale Y mode	Automatic	the Y scale of the FFT diagram is adapted automatically
	Manual	the Y scale of the FFT diagram can be specified manually in the toolbox that appears - the Y scale can also be specified by clicking directly on the values on the Y-axis
Result Unit	$Unit_{RMS}$	RMS of selected oscilloscope unit
	$Unit_{PK}$	peak of selected oscilloscope unit
	$Unit_{RMS}^2$	RMS power of selected oscilloscope unit
	$Unit_{PK}^2$	peak power of selected oscilloscope unit
	$Unit_{RMS}/\sqrt{Hz}$	RMS density of selected oscilloscope unit
	$Unit_{PK}/\sqrt{Hz}$	peak density of selected oscilloscope unit
	$Unit_{RMS}^2/Hz$	RMS power density of selected oscilloscope unit
	$Unit_{PK}^2/Hz$	peak power density of selected oscilloscope unit

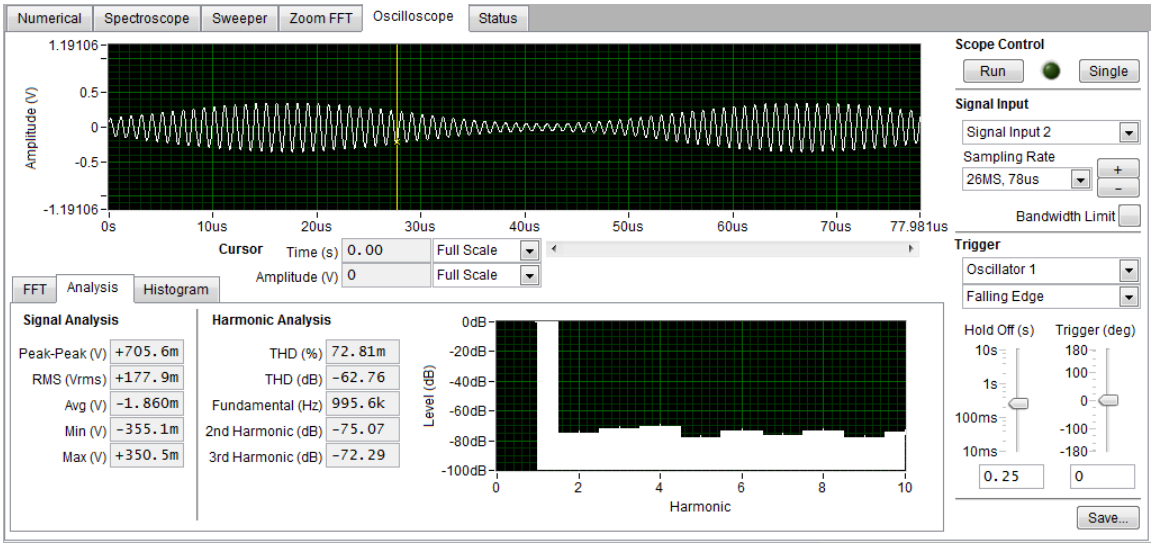


Figure 5.14. Oscilloscope tool with Analysis tab

Table 5.17. Oscilloscope/Analysis tab

Control/Tool	Options/Range	Description
Signal Peak-Peak value	-	maximum to minimum value of the samples in the displayed waveform
Signal RMS value	-	the root-mean-square of the samples in the displayed waveform
Signal Avg field	-	the average value of the samples in the displayed waveform
Signal Min field	-	the minimum value of the samples in the displayed waveform
Signal Max field	-	the maximum value of the samples in the displayed waveform
Total Harmonic Distortion (THD) field	-	the total harmonic distortion of the samples in the displayed waveform
Total Harmonic Distortion (THD) diagram	-	the total harmonic distortion diagram depicted with the harmonics in the X scale

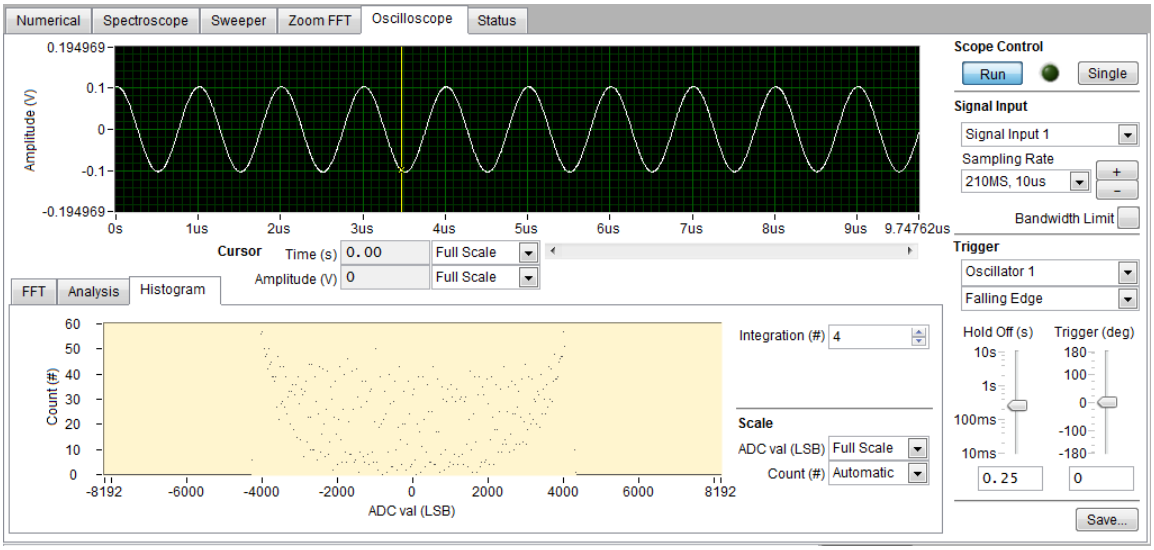


Figure 5.15. Oscilloscope tool with Histogram tab

Table 5.18. Oscilloscope/Histogram tab

Control/Tool	Options/Range	Description
Histogram display	-	shows the number of samples in the displayed waveforms with the value at the converter in the X scale
Integration select	1 to 256	defines the averaging on a number of waveforms, achieving a more stable histogram view
Scale X mode (ADC values)	Manual	makes the toolbox appear in order to set the X scale manually - select one scale label by clicking on it, and enter a new value - accept the new label by clicking somewhere in a neutral area of the GUI
	Full Scale	sets the X scale to full scale, equivalent to the resolution of the input ADC (from -8192 to 8192)
Scale Y mode (count values)	Automatic	sets the Y scale of the histogram display in automatic mode, so that the range is calculated depending on the actual waveform sample count
	Manual	makes the toolbox appear in order to set the Y scale manually - select one scale label by clicking on it, and enter a new value - accept the new label by clicking somewhere in a neutral area of the GUI

5.4.6. Status Tool and History Log

This section describes the features available in the status tab and in the history log section.

Features:

- First level instrument status information for problem solving (4 status indicators)

- Signal input level status, USB communication status
- Command history log

Description:

At the bottom-right area of the screen, 4 status indicators with 2 conditions each show notifications of the current status of the instrument. All status indicators indicate that a problem with the device is currently occurring or has occurred in the past. It is possible to reset the status indicators by pressing the return arrow symbol at the bottom-right. In case the status conditions existed in the past and are not true anymore, all indicators can be cleared at once. The clearing has not effect if the problem condition still exists: inspection of the current measurement conditions is required instead.

Warning

The data acquired when an error condition occurs may be faulty, incomplete, or inconsistent. Users should consider discarding the acquired data when a error condition has occurred during the measurement.

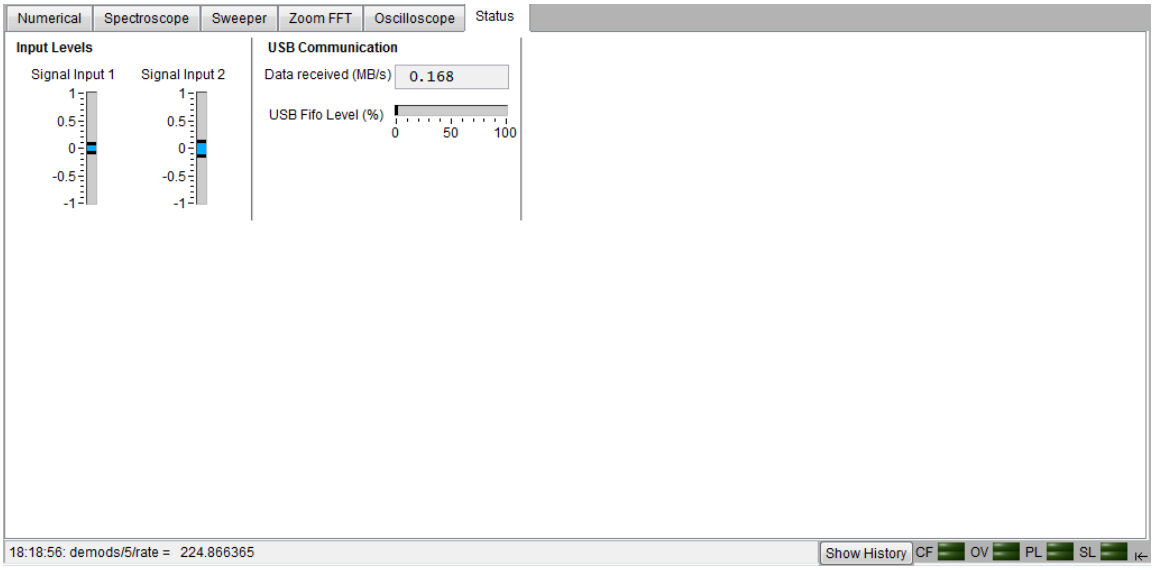


Figure 5.16. Status tool

The history log shows all commands that have been applied to the instrument since the start of the current ziControl session. There is a timestamp when the command has been performed and the command in the same syntax as for the text programming interface or for a LabVIEW program.

When clicking on the status line or on the Show History button, the complete history log opens and can be examined in detail. Copy and pasting of text parts is supported. Click again on the status line or the Hide History button to collapse the log.

The complete sequence of commands since the start of the ziControl session is also stored to a log file in the LabVIEW Data directory which is usually in this location `C:\Users\[USER]\Documents\LabVIEW Data` and the file name is called `com.zhinst.ziControlStatusLog.txt`.

Figure 5.17. History log

Table 5.19. Status tool

Control/Tool	Options/Range	Description
Input signal 1 level	-1 to +1	indicates in graphical format the current level at signal input 1 - the occurrence of an overshoot at signal input 1 lights the Input Over (OV) flag
Input signal 2 level	-1 to +1	indicates in graphical format the current level at signal input 2 - the occurrence of an overshoot at signal input 2 lights the Input Over (OV) flag
USB communication rate	[MB/s]	this level indicates the current communication throughput between the HF2 Instrument and the host computer - the rate is directly related to the readout rate of the demodulators - a level below 7 MB/s is suggested to avoid data loss
USB FIFO level	0% to 100%	the level indicates the current status of the USB FIFO buffer - a level above 100% during a short amount of time is equivalent to a package loss and the Package Loss (PL) flag is set - the FIFO level is dependent on many parameters, but in particular on the host computer and its operating system - higher performance computers are able to keep the FIFO level at a low level
CF (Clock fail)	Status	external clock is selected and does not work
	Occurred	clock fail has occurred in the past
OV (Input over voltage)	Status	the signals at Input 1 and/or Input 2 are overshooting the selected range and clipping of the samples occurs
	Occurred	input over voltage has occurred in the past
PL (Package loss)	Status	the USB communication is experiencing serious overflow and data packages are being discarded (samples and control) - this problem may be related to a high demodulator sample readout setting, however other reasons may also exist

Control/Tool	Options/Range	Description
	Occurred	package loss has occurred in the past
SL (Sample loss)	Status	the USB communication is experiencing serious demodulator sample losses - a sample loss may provoke or contribute to provoke a package loss - sample loss is less serious than package loss, however it may compromise the measurement integrity - a demodulator sample loss can be avoided by reducing the demodulator readout rate
	Occurred	sample loss has occurred in the past
Clear button	[press once]	clears all error flags

Chapter 6. Communication and Connectivity

This chapter describes the different possibilities to interface with an HF2 Instrument. The HF2 Series was designed with the concept that "the computer is the cockpit"; there are no controls on the front panel of the HF2 Instrument, instead the user has the freedom to configure and stream data from the instrument directly from their computer. The aim of this approach is to give the user the freedom to choose where they connect to, and how they control, their HF2 Instrument. The user can connect directly from a computer connected to the HF2 Instrument via USB or remotely from a different computer on the network, away from their experimental setup. Then, on either computer, the user can configure and retrieve data from their HF2 Instrument via a number of different interfaces, i.e. via ziControl and/or their own custom programs. In this way the user can decide which connectivity setup and combination of interfaces best suits their experimental setup and data processing needs.

We first provide an overview of how the user connects an HF2 Instrument to a PC in Section 6.1 and then give an overview of how to quickly modify instrument settings using using the text-based console in Section 6.2 . Finally, at the end of this chapter, we explain how to connect to an HF2 instrument over a public network, Section 6.3 .

Note

It is also possible to configure and obtain data from an HF2 Instrument via one of our APIs. Currently LabVIEW, Matlab, Python or C are available. These topics are covered in a separate document, The LabOne Programming Manual.

Note

New users could benefit by first familiarizing themselves with the instrument using the graphical user interface ziControl, see Chapter 3 .

Note

Programming using the Real-time Option (RTK) is dealt with in Section 8.2 .

6.1. Instrument Connectivity Overview

The HF2 Series supports a server-based connectivity methodology for multi-user, multi-device operation. This means that it is possible to operate more than one HF2 Instrument from a single computer, that multiple users may access the same instrument, and that an instrument may be made available on a local area network. Server-based means that all communication between the user and the HF2 is via a computer program called a server, in our case ziServer. The ziServer program recognizes the device and manages all communication between the instrument and the host computer over the USB connection on one side, and the different available interfaces on the other side.

Before going into more detail, the terminology used in this chapter is explained.

- Host computer: The computer that is directly connected to the HF2 by USB. An HF2 can only be connected to one host computer, but to multiple remote computers on a local area network via ziServer running on the host.
- ziServer : A computer program that runs on the host computer and manages settings on, and data transfer to and from the HF2 by receiving commands from clients. It always has the most up-to-date configuration of the device and ensures that the configuration is synchronized between different clients.
- Remote computer: A computer, available on the same network as the host computer, that can communicate with the HF2 via the ziServer program running on the host.
- Client: A computer program that communicates with the HF2 via the server. The client can be running either on the host or the remote computer.
- API (Application Programming Interface): a collection of functions and data structures which enable communication between software components. In our case, the various APIs (e.g., LabVIEW, MATLAB®) for the HF2 provide functions to configure the device and receive measured experimental data.
- Interface: Either a client or an API.
- TCP/IP: Network communication protocols. In our case, ziServer communicates to the base API (ziAPI) using TCP/IP. This can happen either locally (entirely on the host computer) or between the host computer and remote computers.
- GUI (Graphical User Interface): A computer program that the user can operate via images as opposed to text-based commands.
- Modules: Software components that provide a unified interface to APIs to perform high-level common tasks such as sweeping data.

An overview of HF2 Instrument connectivity is shown in Figure 6.1 .

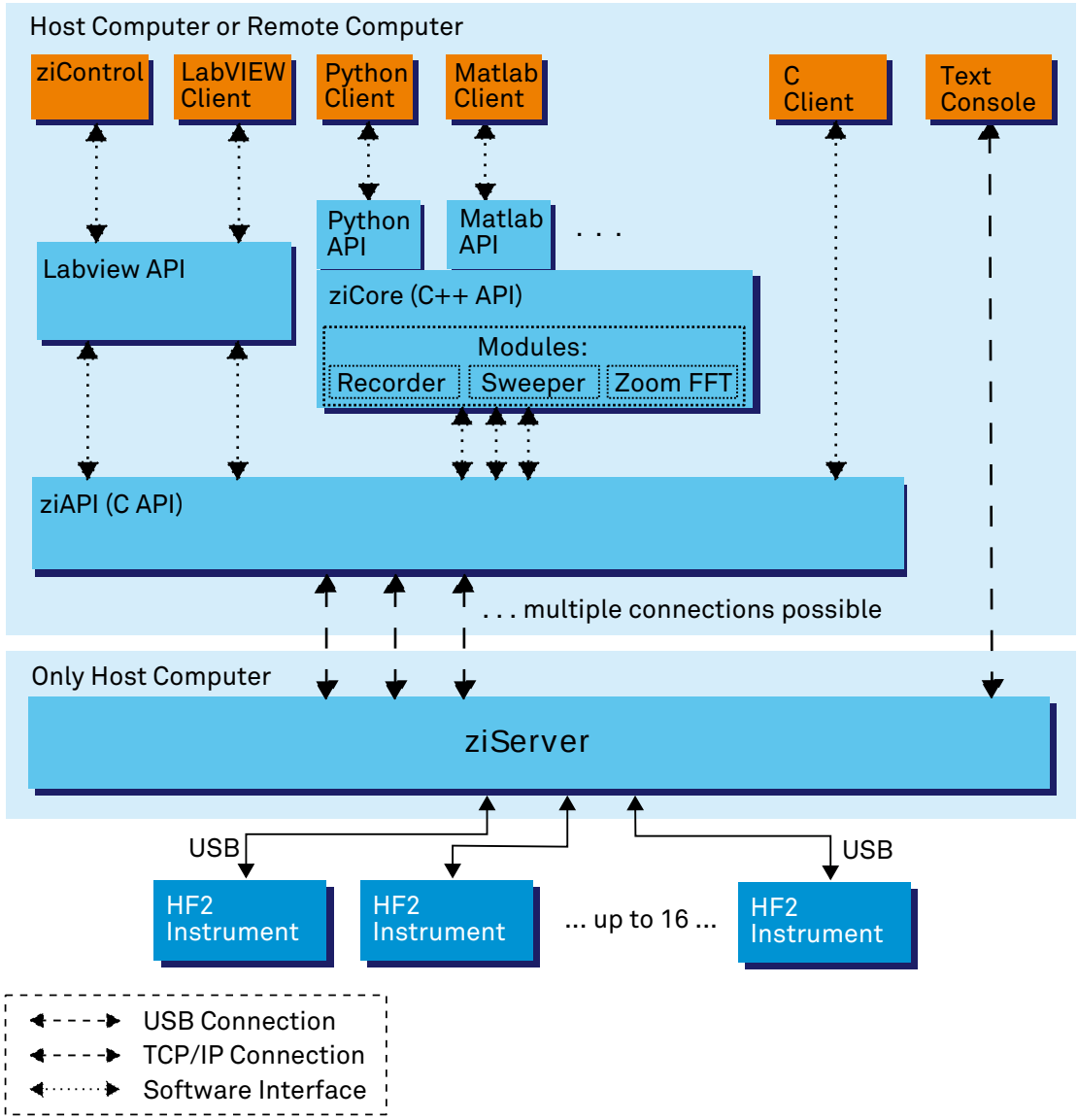


Figure 6.1. Instrument Connectivity

6.1.1. Physical Connectivity: Host and Remote Computers

In a commonly used configuration, the HF2 Instrument is connected to a host computer where both the server and the clients (denoted by the orange boxes in Figure 6.1) run. The ziServer program acts as bridge from the instrument to the various clients. For example, a user may use two clients in parallel: ziControl to configure the device and their own program created using the LabVIEW API to plot custom results streamed from the instrument. Both these clients communicate data via the same instance of ziServer and ziServer ensures that both clients are always updated with the current instrument configuration. Note however, that any combination of clients shown in Figure 6.1 may be used in parallel, limited only by the performance of the host computer and by the load from requests to ziServer. In this configuration, the top and bottom block of Figure 6.1 (denoted by the light blue box) are both running on the host computer.

Sometimes, the user wishes to use a client to control the HF2 on a remote computer. In this case, the software in the top block of Figure 6.1 runs on the remote computer, connecting via TCP/IP over the local area network to the instance of ziServer running on the host computer (which is connected to the HF2 via USB).

In total, there are three possibilities of physically connecting to an HF2 Instrument:

- On the host computer, i.e., all the software (ziServer, interfaces) is running on the same computer that is connected to the instrument via USB. This is the simplest and most common setup.
- On a remote computer connected to the host computer over a secure local area network. If a private network is available this is a simple setup, ziServer only needs to allow remote connections, see the section called “Enabling a Remote Connection to ziServer” .
- On a remote computer connected to the host computer over a public, insecure network. This is a more advanced topic covered in Section 6.3 .

As you can now imagine, there are many possibilities to connect to an HF2 instrument. The following methods of connecting with HF2 Instruments are supported:

- Connection to and operation of an HF2 Instrument from multiple clients on different computers in parallel with automatic background update of instrument settings on all connected clients.
- Connection to and operation of up to 16 HF2 Instruments from a single host computer.
- Connection to and Operation of multiple remote HF2 Instruments that are connected on a TCP/IP LAN via a (or multiple) host computer(s), the number of which is limited by the performance of the remote computer. Note, there can only be one instance of ziServer running for one HF2 Instrument.

6.1.2. Software Connectivity: ziServer

The ziServer program provides a gateway to your HF2 Instrument from any of the programming interfaces described in this chapter. The ziServer program recognizes the device and manages all communication between the instrument and the host computer over the USB connection on one side, and the different available interfaces on the other side. Since ziServer is responsible for all communication to the instrument, it's important that only one instance of ziServer is running at any one time. This is how you can check that only one instance of ziServer is running, or is indeed running at all:

- Windows: Open Windows Task Manager with Ctrl-Shift-Esc and check that both the processes `ziServer.exe` and `ziService.exe` are running.
- Linux: Either check manually that the process `ziServer` is running or alternatively use the `ziService` command

  ```
  $ ziService status
  ```

  in a terminal. You should see the output:

  ```
  Status : ziServer is running.
  ```

Enabling a Remote Connection to ziServer

In order to enable connections to ziServer from a remote computer, the node `/zi/config/open` must be set to `1`. To set this in ziControl go to the "Connectivity Tab" and under the "Connectivity to ziServer" section enable, "From Anywhere" in the "Connectivity" setting.

6.1.3. Instrument Communication: The Node Hierarchy

In order to communicate with an HF2 Instrument via text-based commands, it is necessary to understand how the settings and measurement data of the instrument are accessed. All settings of the HF2 Instrument are organized in a file-system-like hierarchical structure. This means that it is possible to plot a consistent tree of nodes, where the instrument settings are leaves of the tree. It is also possible to browse branches inside the tree as if the user were navigating in a file-system. This hierarchy is used, no matter which interface you use when performing measurements.

An example demonstrating the hierarchy is the representation of the first demodulator on the device, given by the node:

```
/devX/demods/0
```

which, as we've already noted, is very similar to a **path** on a computer's file-system. Note that, the top level of the path is the device that you are connected to. The demodulators are then given as a top-level **node** under your device-node and the node of the first demodulator is indexed by `0`. This path represents a branch in the node hierarchy which, in this case, if we explore further, has the following nodes:

```
/devX/demods/0/adcselect
/devX/demods/0/order
/devX/demods/0/timeconstant
/devX/demods/0/rate
/devX/demods/0/trigger
/devX/demods/0/oscselect
/devX/demods/0/harmonic
/devX/demods/0/phaseshift
/devX/demods/0/sinc
/devX/demods/0/sample
```

These nodes are **leaves** , the most bottom-level nodes which represent a setting of an instrument or a field that can be read to retrieve measurement data. For example, `/devX/demods/0/adcselect` is the leaf that controls the setting corresponding to the choice of signal input for the first demodulator. To set the index of the signal input the user writes to this node. The leaf `/devX/demods/0/sample` is the leaf where the demodulator's output (timestamp, demodulated x-value, demodulated y-value) are written at the frequency specified by `/devX/demods/0/rate`. In order to obtain the demodulator output you read the values from this node by **polling** this node. Polling a node sends a request from the client to ziServer to obtain the data from the node at that particular point in time.

Chapter 7 provides a full reference of nodes on HF2 Instruments and details which settings or measurement data they correspond to, whether they are read-only and, if they are writable, which values they may take (e.g., boolean, integer, floating point).

Note

The numbering on the front panel of the HF2 Instrument and the block numbering on the graphical user interface ziControl generally start with 1, whereas the underlying instrument using the programming interfaces has a numbering notation starting with 0.

Note

A useful method to learn about paths in your HF2 Instrument is to look at the output of the history in the bottom of the graphical user interface ziControl. The status line always shows the last applied command and you can view the entire history by clicking the 'Show History' button. You will find paths like

```
/devx/sigins/0/ac = 1
```

after you switched on the AC mode for signal input 1, or

```
/devx/demods/1/rate =
        7200.000000
```

after setting the readout rate of demodulator 2 to 7.2kHz.

You can obtain a list of nodes available on your instrument as a text-file in ziControl by saving the instrument settings. Go to the "Save" tab in ziControl and in the "Save/Load HF2 Settings" click the "Save Settings..." button.

Note

We recommend that users who want to program their HF2 Instruments first familiarize themselves with the node hierarchy by browsing nodes via ziServer's text-based interface described in the next chapter. The text-based interface is an indispensable tool for HF2 programmers.

6.2. ziServer's Text-based Interface

The text-based interface is the simplest and most direct way of communicating with an HF2 Instrument and doesn't require any previous programming experience. Browsing the text interface physically happens within ziServer and since it makes use of TCP/IP sockets the user can also connect remotely over a network connection via telnet or ssh. In contrast to ziControl, this is a geeky way of using an HF2 Instrument.

After connecting to the text-based interface via telnet, you find yourself in a DOS or Unix terminal-like program, where you can browse instrument settings in the node hierarchy (Section 6.1.3). The terminal responds to known command syntax like `ls` (list all nodes in the current directory) and `cd` (select path to navigate in the directory hierarchy).

The text interface is a very powerful tool for users programming an HF2 Instrument with other interfaces such as Zurich Instrument's LabVIEW or MATLAB® API. It is a convenient way to verify the instrument's node paths and check that values have been set correctly by the interface you are actually programming with. It is also helpful for budding HF2 hackers who can use it to browse the node hierarchy and familiarize themselves with its structure.

Note

In theory, it would be possible to use the text-based interface to communicate with an HF2 Instrument from an arbitrary programing environment. However, this would require the implementation of a socket connection and a parser, and there is no exception handling should a command fail. Also, since its a text interface, as opposed to a binary interface, data transfer is slower. Therefore, in general, we strongly encourage the user to instead use one of the existing binary interfaces documented later in this chapter as their primary programming interface.

6.2.1. Getting Started with the Text-based Interface

Preparation

The purpose of this section is to get quickly acquainted with the text interface to the ziServer. For this you will need to have installed LabOne (see Section 1.2) and have your HF2 Instrument connected to your host computer via USB. In order to access the text-based interface within ziServer, a telnet or SSH client providing a console is required.

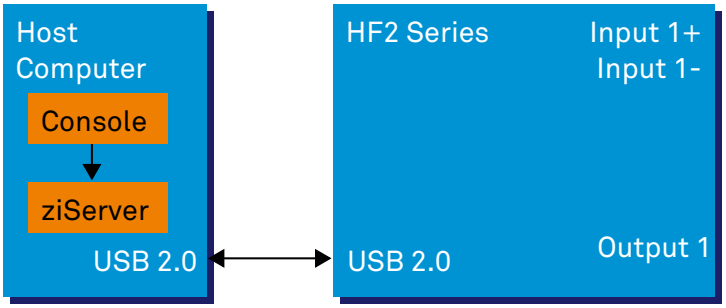


Figure 6.2. Setup for using the text-based interface

Connecting to ziServer on Windows

Zurich Instruments recommends to use the freeware PuTTY as a telnet client. PuTTY has to be configured with the following settings to connect with ziServer.

Table 6.1. PuTTY settings on Windows

Terminal category, Implicit CR in every LF	set
Session category, Host Name	localhost
Session category, Port	8005
Session category, Connection type	Telnet

Users connecting to a remote ziServer (a ziServer which is not running on the local machine, but on the host computer available on the LAN) have to configure the host name accordingly (e.g. computer.domain.com) after allowing remote connections to ziServer, see Section 6.1.2 .

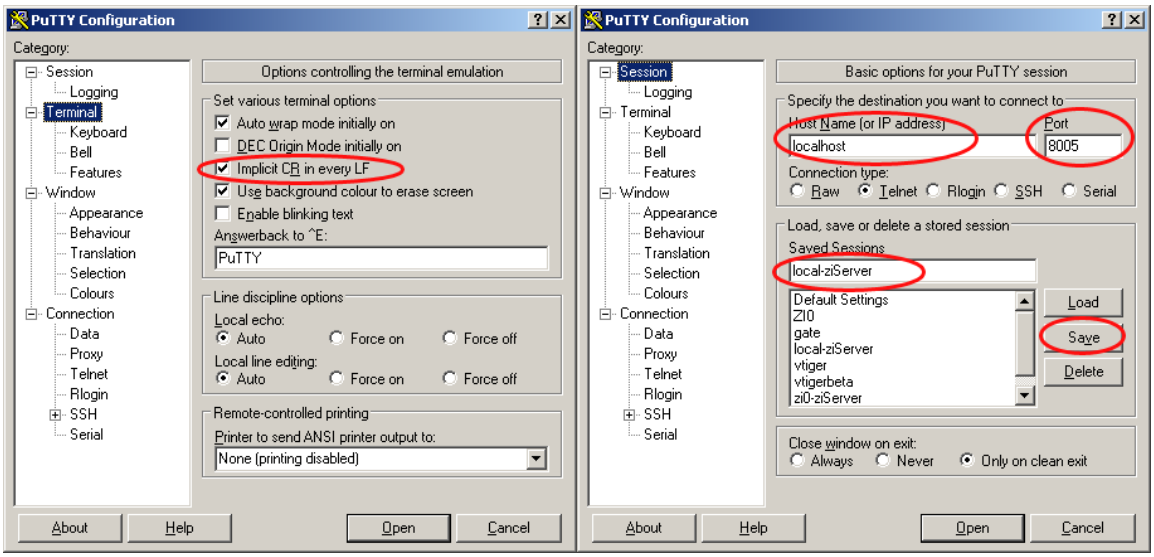


Figure 6.3. PuTTY configuration to connect to ziServer

Save the session settings with a suitable name, so that you can connect faster next time. After pressing the Open button, the following screen will appear: this message confirms successful connection to the ziServer. If the screen does not appear, or the text is missing, please check whether ziServer is running (Windows task manager, see Section 6.1.2) or check your PuTTY settings.

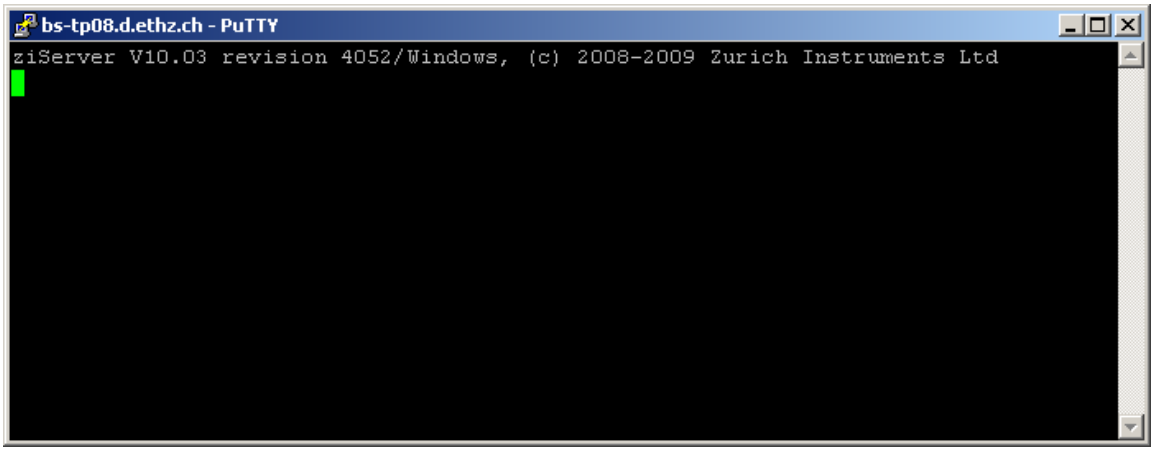


Figure 6.4. PuTTY successful ziServer connection

Connecting to ziServer on Linux

You may connect to a running ziServer from the host computer by invoking telnet in a shell:

```
            user@zi:~$ telnet localhost 8005
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
ziServer V14.08 revision 25429/Linux, (c) 2008-2014 Zurich Instruments AG
```

Or by using netcat:

```
            user@zi:~$ nc localhost 8005
ziServer V14.08 revision 25429/Linux, (c) 2008-2014 Zurich Instruments AG
```

A Tour of the Text-based Interface

We start our tour with some basic commands. After successful connection, it's nice to see which instruments are connected to ziServer. An `ls` will do the job. This yields the information that we have a ZI node (the node for ziServer) and a DEVX node (denoting your HF2 Instrument). The DEVX is the serial number of the HF2 Instrument in front of you. Let's select the ziServer node with `cd zi`, list the nodes with `ls`, and then read all values of the node inside the /ZI/ tree with `*/* ?`. Not very impressive so far.

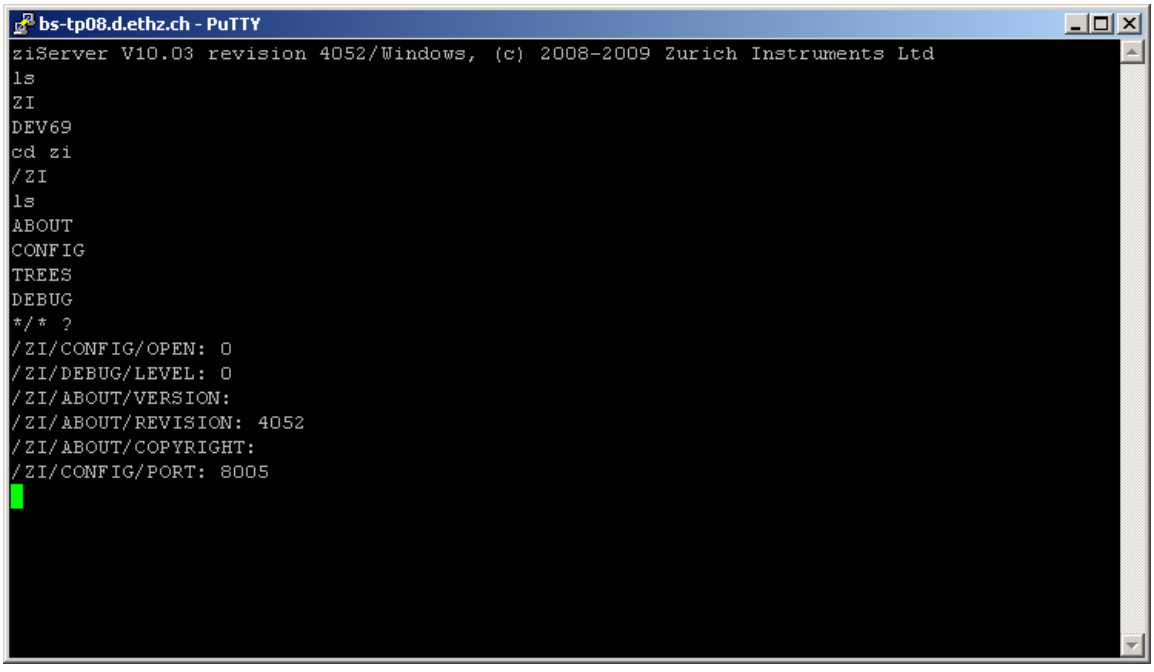


Figure 6.5. PuTTY tour: check server version

Let us move into the DEVX hierarchy by using the relative path `cd ../devx` (it's also possible to specify absolute paths, e.g., `cd /devx/` and investigate the structure of the node hierarchy with

the `ls` command. This lists all the leaves inside of your device. Each leaf represents a setting that can be made inside of the instrument or a field that can be read to retrieve measurement data. The first level hierarchy inside the instrument is displayed in Figure 6.6 .

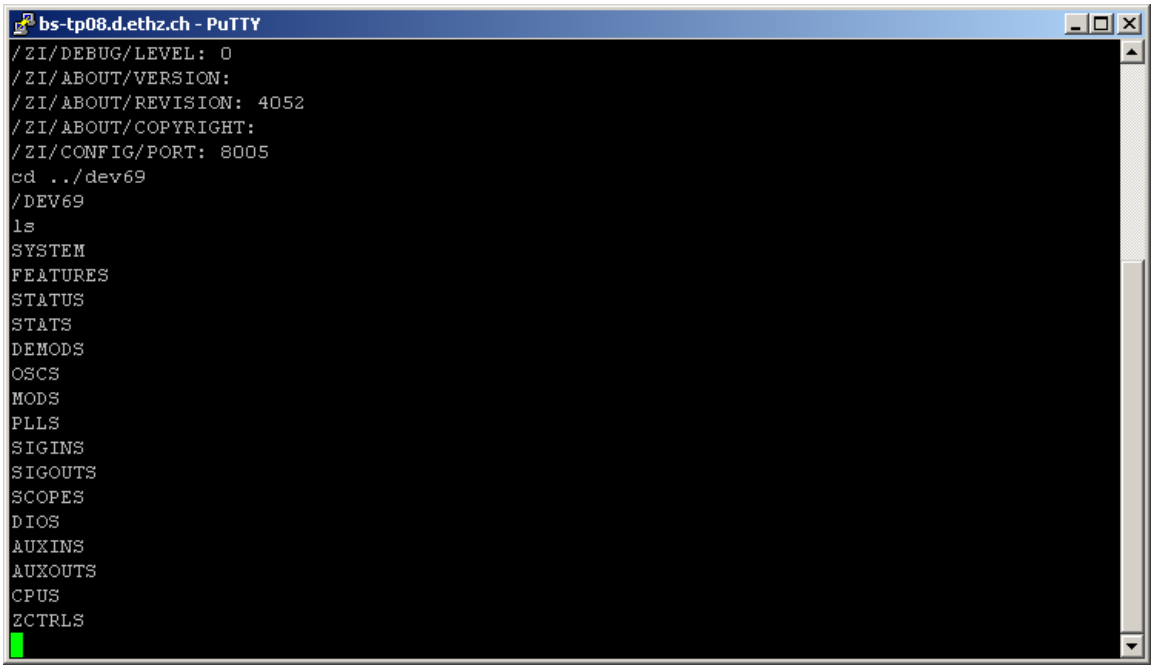


Figure 6.6. PuTTY tour: first instrument hierarchy

This list gives a top-level insight into an HF2 Instrument showing its building blocks such as DEMODS (demodulators), OSCS (oscillators), SIGINS (signal inputs), SIGOUTS (signal outputs), SCOPES (oscilloscopes), AUXINS (auxiliary inputs), AUXOUTS (auxiliary outputs), CPUS (integrated processors), and so on. The branches and leaves that you see will depend on the options installed in your device: for instance, you will not see PLLS if you do not have the HF2PLL option installed.

It is time to dive into one branch of the instrument. Let us take oscillator 0: type `cd oscs`, then `ls` to see the branches at that level, then type `cd 0` to select the first oscillator, then list the leaves at that level, and use `* ?` to return the values of all leaves. We see for instance that `/DEVX/OSCS/0/FREQ` has a value of 2.5 MHz, see Figure 6.7 .

It is now possible to check that ziControl actually has the same value in the corresponding field. Note, that the block numbering notation inside of the GUI starts with 1, whereas the underlying instrument has a numbering notation starting with 0. It is also possible to change the frequency of the lock-in channel 1 inside of the GUI to 2.1 MHz, and then check the value inside the text interface by typing `* ?`. You notice that the settings changes are transparent to all clients connected to a ziServer. You can always rely on setting and data consistency.

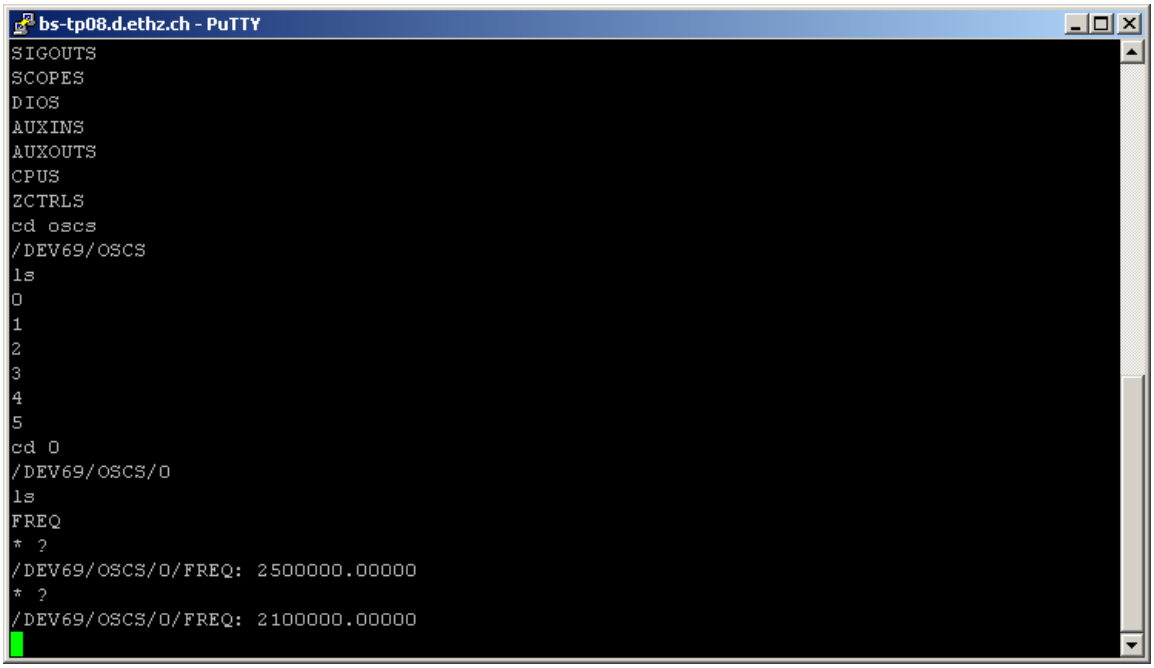


Figure 6.7. PuTTY tour: leaves of an oscillator

Next, to change the value of the oscillator frequency, for instance to 4.5 MHz, type `freq 4500000`. The same effect can be achieved by using the absolute path `/DEV8/OSCS/0/FREQ 4500000`. Please note that the value in the GUI has changed from 2.1 MHz to 4.5 MHz in the meantime.

The wildcard symbol * can simplify life when many similar settings need to be made. Lets for instance check the frequency of all oscillators at once: type `cd ..`, and then `*/freq ?`, and then change all frequencies to 3.6 MHz with `*/freq 3600000`. This is where the text interface is becoming pretty powerful.

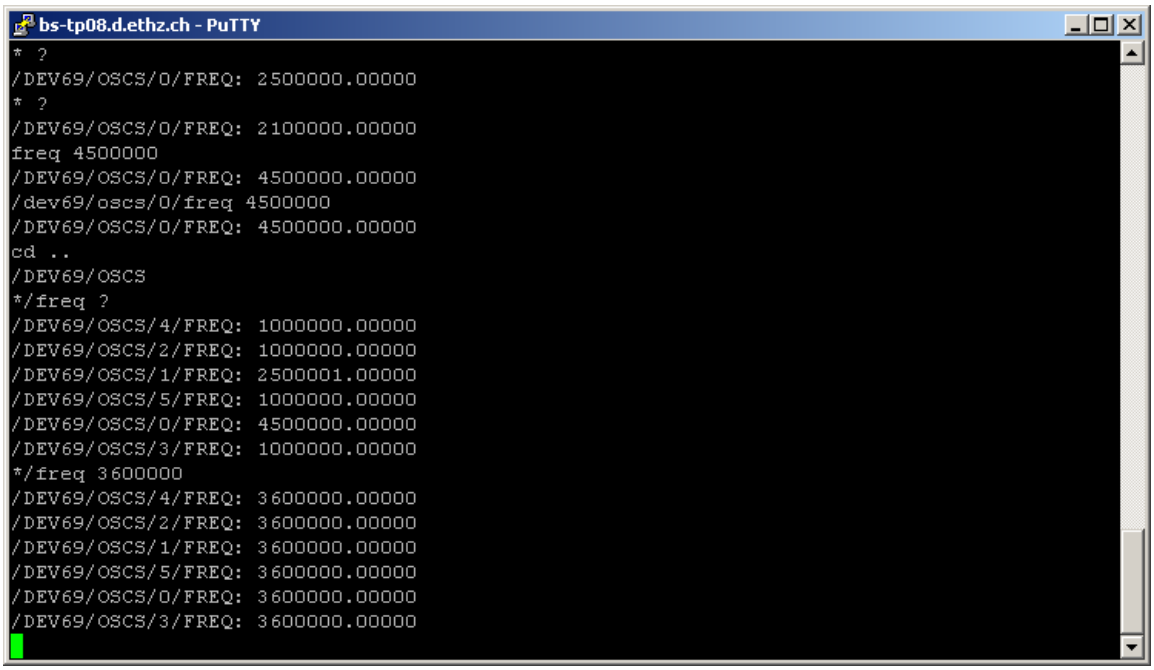


Figure 6.8. PuTTY tour: using the wildcard symbol

One word on scripting. It is possible to manually compile several settings in a file using the syntax `path value`, then to copy-paste them into the terminal window. The sequence will be recognized by the ziServer and all defined settings will be made.

Note, another useful method to learn about the paths in your HF2 Instrument is to look at the bottom of the ziControl after changing configuration (see this Note). The complete command history of a session is stored in the `LabVIEW Data` directory, file `com.zhinst.ziControlStatusLog.txt`.

This concludes getting started with text-based programming. Zurich Instruments hopes you found it useful, and hopes you are going to perform some tutorials in Chapter 3 . Thank you for measuring with Zurich Instruments.

6.2.2. Command Reference

Commands

The following lists all available commands in the text-based interface, it can be viewed in the interface by typing `help`.

```
            Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
ziServer V14.08 revision 25429/Linux, (c) 2008-2014 Zurich Instruments AG

help

ziServer V14.08 revision 25429/Linux, (c) 2008-2014 Zurich Instruments AG

All parameters from and to the device are organized in a tree structure. This simple
 interface allows you to interact with the device and use all its features based on
 an ascii-protocol.

The interface holds a current working node to which all paths you give are relative
 unless you begin a path with a "/". You may use ".." to reference a node one level
 higher in hierarchy. In all paths wildcards "*" may be used.

The following commands are recognized by the ascii-interface:

[path] [value] Sets the value of the node(s) at the specified path.
[path] ? Gets the value of the node(s) at the specified path.
sel|cd [path] Sets the working path according to the provided one.
sel|cd ? Prints the current working path.
ls [path] Lists all children available in the given path.
tr [path] Prints all children and children's children of the specified path as a
 tree.
subs [path] Subscribes the nodes of the given path for event forwarding.
unsubs [path] Unsubscribes the nodes of the given path from event forwarding.
exit  terminates the running session
help  Shows this page
```

Note

The text-based interface is case insensitive.

Nodes, Leaves and Paths

Every setting of the instrument is represented by a leaf as a terminal of a tree of nodes. There are also leaves which are not settings, but for instance used to retrieve data from the instrument. For each leaf there is a path and the related value.

```
path_list = path [path]
path = [/|/..|*]name[/name|*|**]
```

In the syntax above a name is a string, and the path is a list of names separated by a slash. If a path starts with a slash, it is an absolute path starting at the root of the hierarchy. The asterisk is a wildcard meaning all nodes at a given hierarchy, and two points in a row means one hierarchy higher.

Navigation and Trees

The navigation inside the text interface is performed with the `sel/cd/ls/tr` commands.

```
sel or cd [?|..|path]
ls [path]
tr [path]
info [path]
```

The command `cd ?` feedbacks the current path, `cd ..` moves up one tree level, `cd path` moves down one tree level. `sel` and `cd` are equivalent commands. `ls` lists the tree available on the current path, `ls path` lists the tree available on the specified path, `tr` lists the complete tree on the current path, `tr path` lists the complete tree on the specified path, `info` feedbacks the help string of the current path, and `info path` reports the help string of a given path.

Get and Set Node Values

The values of nodes are read and changed with the following syntax.

```
path ?
path_list value
```

The command `path ?` returns the value of path, `path value` sets the specified node to value, and `path_list` value sets several nodes to value. Some examples:

```
about/* ?             // return values of leaves at path
devx/demods/0/* ?
/zi/config/* ?

/devx/demods/0/adcselect ?     // return value at path
/devx/demods/0/adcselect 0     // set value of leaf

/devx/demods/0/adcselect /devx/demods/1/adcselect 1
                               // multiple set value
```

Subscriptions

The ziServer provides a mechanism to automatically send all changes to a leaf to a subscribed client. This mechanism efficiently informs a client whenever a setting or a data of the instrument has changed without the need of active polling. It is possible to subscribe to single leaves, or full trees.

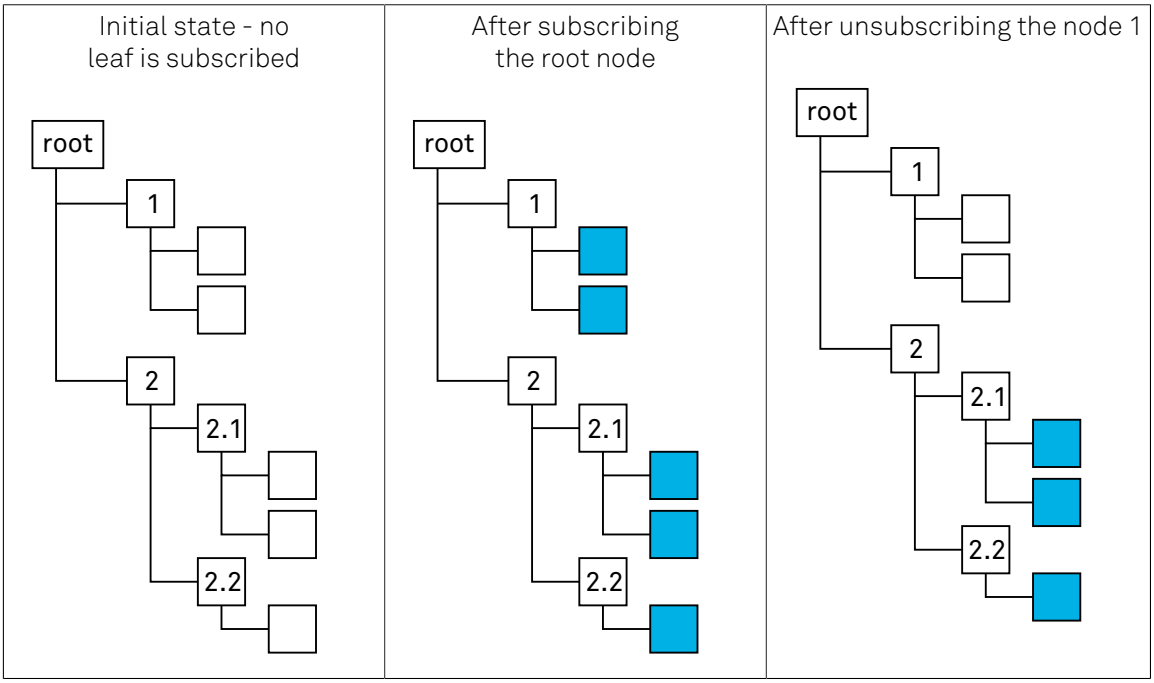
When a value of a subscribed leaf changes, the updated value is sent to the client. Most often samples, error and status nodes are subscribed. If one needs to maintain a user interface, then this can be done using subscriptions.

If you subscribe or unsubscribe from a node which is not a leaf, the subscription propagates to all nodes of the subtree. For example, you could first subscribe a subtree and then unsubscribe specific nodes within this subtree and still receive events for all nodes except for unsubscribed ones.

```
subs path_list       // subscribe
unsubs path_list     // unsubscribe

path value           // return value for subscribed leaf
```

The following sequence illustrates subscribe and unsubscribe commands following each other, where turquoise leafs denote subscribed leafs.



The first image shows the initial state with no leaf subscribed. This state corresponds to a newly initiated ziServer session. After subscribing the root node, all leafs become subscribed. Then it is for instance possible to unsubscribe node 1 in order to leave just the leafs below node 2 subscribed.

Scripting

It is possible prepare a sequence of commands in a text editor and copy-paste them into the terminal session. The console will send all commands to the ziServer and the server will interpret them one by one.

Note

Use the right-mouse button in order to copy-paste into a Windows Putty session.

```
cd /DEVX
SIGOUTS/*/ON 0
SIGOUTS/0/RANGE 1
SIGOUTS/0/AMPLITUDES/0 1
SIGOUTS/0/ENABLES/* 0
SIGOUTS/0/ENABLES/0 1
OSCS/0/FREQ 300000
SIGINS/0/RANGE 10
DEMODS/0/BITSHIFT 8
DEMODS/0/ORDER 2
DEMODS/0/RATE 1000
```

6.3. Connecting to ziServer over insecure or firewalled networks

If you want to connect to the ziServer over insecure, public networks like the public internet, you need to consider that the TCP/IP connection to the ziServer is unsecured. Also many firewalls will not allow traffic to port 8005. There are two common solutions to this problem. Either a VPN or a ssh port tunneling/forwarding. In this section ssh port tunneling/forwarding is described.

6.3.1. SSH port forwarding

You can use ssh to connect to a remote computer and use this connection to tunnel ziServer traffic between the local and remote computer.

To illustrate how port forwarding works, let us use an example. Suppose you have two buildings. In Lab #1, there is the lab with computers residing in the subnet 10.1.1.* and the HF2 is connected to one of these computers. At your Home, there are office computers residing in the subnet 10.2.2.*. The computers in Lab #1 are running the ziServer application that uses an unencrypted TCP/IP session to communicate data with, e.g., ziControl at your home. The firewall of the Lab and your Home might not permit this connection to be initiated. There are two kinds of port forwarding: local and remote forwarding. They are also called outgoing and incoming tunnels, respectively. Local port forwarding forwards traffic coming to a local port to a specified remote port. For example, all traffic coming to port 1234 on the client could be forwarded to port 8005 on the server (host).

The value of localhost is resolved after the Secure Shell connection has been established – so when defining local forwarding (outgoing tunnels), localhost refers to the server (remote host computer) you have connected to. Remote port forwarding does the opposite: it forwards traffic coming to a remote port to a specified local port. For example, all traffic coming to port 1234 on the server (host) could be forwarded to port 8005 on the client (localhost).

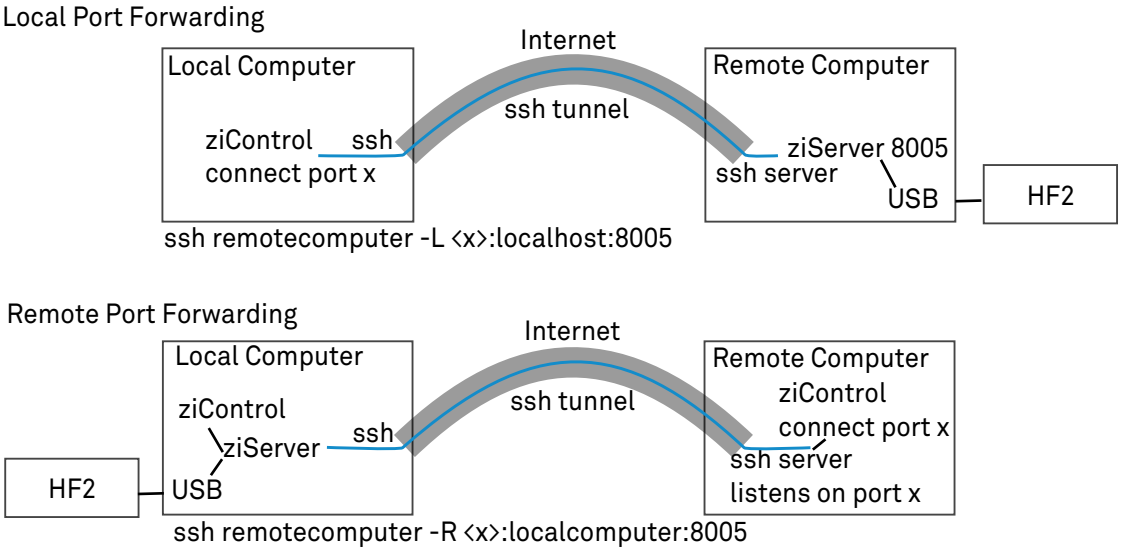


Figure 6.9. Secure connectivity

6.3.2. Local port forwarding

Accessing a service (in this example ziServer port tcp/8005) on a machine in the laboratory (10.1.1.*) from your machine at home (10.2.2.*), simply by connecting to the server `work.example.org` at work:

```
$ssh user@work.example.org -L 10000:172.16.10.10:8005
```

We see the ziServer is available on the loop back interface only, listening on port tcp/10000:

```
$ netstat -tunelp | grep 10000

tcp 0 0 127.0.0.1:10000 0.0.0.0:* LISTEN 1000 71679 12468/ssh
```

From your home machine, you should be able to connect to the machine at work:

```
$ telnet localhost 10000
```

By specifying localhost and port 10000 in ziControl you can connect with ziControl. Note that port 10000 is chosen arbitrarily.

6.3.3. Local port forward for anyone at home

If you want other people on your home subnet to be able to reach the machine at work by SSH, add the global option -g:

```
$ ssh user@work.example.org -L 10000:172.16.10.10:22 -g
```

We now see the service is available on all interfaces on your home computer (10.2.2.5), available for anyone to connect to on the local subnet:

```
$ netstat -tunelp | grep 10000 tcp 0 0 0.0.0.0:10000 0.0.0.0:* LISTEN
1000 72265 12543/ssh
```

Anyone on your local subnet should be able to connect to the machine at work by doing this:

```
$ telnet 10.2.2.5 10000
```

By specifying host 10.2.2.5 and port 10000 in ziControl you can connect with ziControl.

6.3.4. Remote port forwarding

Giving access to a ziServer (port tcp/8005) on your home machine (10.2.2.5) to people at work:

```
$ ssh user@work.example.org -R 10000:10.2.2.5:8005
```

We see on our server at work (on the loop back interface on port tcp/10000) that we have access to our SSH server at home:

```
work.example.org$ netstat -tunelp | grep 10000 tcp 0 0 127.0.0.1:10000
0.0.0.0:* LISTEN 0 73719534 3809/1
```

People logged in on the machine `work.example.org` now should be able to SSH into your home machine by doing:

```
work.example.org$ telnet localhost 10000
```

6.3.5. Remote port forwarding for anyone at work

If you want everybody on the subnet at work to be able to SSH into your home machine, there is no -g option for remote forward, so you need to change the SSH configuration of `work.example.org`, add to `sshd_config`:

```
GatewayPorts yes
```

Connect just as before:

```
home$ ssh user@work.example.org -R 10000:10.2.2.5:8005
```

Now, it is listening on all interfaces on the server at work:

```
work.example.org$ netstat -tunelp | grep 10000 tcp 0 0 0.0.0.0:10000
0.0.0.0:* LISTEN 0 73721060 4426/1
```

Anyone at work can now connect to your home machine by SSH via the server:

```
anyone.example.org$ telnet work.example.org 10000
```

Chapter 7. Node Definitions

This chapter provides full information about the HF2 Instruments' settings and output. All settings (e.g., demodulator filter order) and results (e.g., demodulator output) are available to the user in a hierarchical tree structure which can be read from and written to. Having the settings of the instrument available to the user in a tree-like structure allows the user to program the instrument for readily repeating complicated experimental setups.

An overview of the node tree structure is provided in Section 7.1 and detailed information about each node, such as whether it's readable or writable, is provided and in Section 7.2 .

Note

For an introduction to the node hierarchy see Section 6.1.3 and to get familiar with reading from and writing to nodes in ziServer's text-based interface see Section 6.2.1 . See also the introduction to Chapter 6 for an overview of the powerful means of programming the HF2 Instrument.

7.1. Overview

Table 7.1. Overview of the Nodetree

Node	Short Description
ZI	This node represents the instance of the ziServer your are connected to.
ABOUT	Node containing information about the server you are connected to.
VERSION	The version of this program.
REVISION	The revision of this program.
FWREVISION	The revision of the used firmware.
COPYRIGHT	The copyright string of this program.
CONFIG	Configuration data of the current instance of the server.
PORT	Configures the TCP/IP port on which the ziServer listens.
OPEN	Configures whether the ziServer should be open for connections from outside the local host.
TREES	Messages on tree changes.
CLOCKBASE	Provides timebase value for the server nodes
DEV0...n	This node represents a single device connected to the server.
CLOCKBASE	Provides clockbase value for the device
SYSTEM	Nodes providing system information and settings.
EXTCLK	Boolean value switching from internal to external clock.
HWREVISION	The revision of the mainboard.
FEATURES	Node containing information on features of the device.
SERIAL	Node providing the serial number of the device.
DEVTYPE	Node providing a string about the type of device.
OPTIONS	Node giving information on enabled options.
CODE	Node providing a mechanism to write feature codes.
STATUS	Nodes providing status information from the device.
TIME	The current timestamp.
FLAGS	Node containing some status flags.
BINARY	A binary representation of all flags.

Node	Short Description
PLLLOCK	Flag indicating if the internal PLL for clock generation has locked.
DCMLOCK	Flag indicating if the internal digital clock manager (DCM) has locked.
FX2RX	Flag indicating if the device receives data via USB.
PKGLOSS	Flag indicating that the device lost data when sending via USB.
MIXERCLIP	Flags indicating that the internal mixer is clipping.
0...n	Flag indicating that this mixer-channel is clipping.
ADCCLIP	Flags indicating that the AD-converter is clipping.
0...n	Flag indicating that this ADC-channel is clipping.
SCOPESKIPPED	Flag indicating that scope data has been skipped.
DEMODSAMPLELOSS	Flag indicating that demod data has been lost.
FIFOLEVEL	Percentage of tx-fifo used.
ADC0MIN	The minimum value on Signal Input 1 (ADC0) during 100 ms.
ADC0MAX	The maximum value on Signal Input 1 (ADC0) during 100 ms.
ADC1MIN	The minimum value on Signal Input 2 (ADC1) during 100 ms.
ADC1MAX	The maximum value on Signal Input 2 (ADC1) during 100 ms.
STATS	Nodes providing statistical data about the device.
BYTESSENT	Total amount of bytes sent via USB.
BYTESRECEIVED	Total amount of bytes received via USB.
MEANPOLLCNT	Average poll-count.
MEANMSGCNT	Average message-count.
PHYSICAL	Group of nodes providing some physical information on the device.
1V2	Actual voltage of the 1.2volts supply.
1V8	Actual voltage of the 1.8volts supply.
2V5	Actual voltage of the 2.5volts supply.
3V3	Actual voltage of the 3.3volts supply.
5V0	Actual voltage of the 5.0volts supply.
TEMP	Actual temperature.

Node	Short Description
DEMODS	Demodulator nodes.
0...n	Nodes of a single demodulator
ADCSELECT	Selects the index of the signal input for the demodulator.
ORDER	Selects the order of the low-pass filter.
TIMECONSTANT	Sets the time constant of the low-pass filter.
RATE	The number of output values sent to the computer per second.
TRIGGER	Sets the trigger- and gating-functionality of the demodulator.
OSCSELECT	Index of the oscillator used to demodulate the signal.
HARMONIC	The harmonic of the base frequency to be used.
PHASESHIFT	Phaseshift of the demodulator.
SINC	Boolean value enabling Sinc filter functionality.
SAMPLE	Samples of the demodulator are given out at this node.
OSCS	Oscillator nodes.
0...n	Nodes of an oscillator.
FREQ	Frequency to of the oscillator.
MODS	Modulator option nodes.
0...n	Nodes of a Modulator.
FMCOEFF	Frequency Modulation Coefficient.
MODE	Modulation mode.
SIDEBAND	Sideband selector.
SAMPLE	Modulation Samples.
PLLS	PLL nodes.
0...n	Nodes of a PLL.
ADCSELECT	Selects an input for the PLL.
AUTOCENTER	Switches autocenter.
FREQCENTER	Selects a center frequency.
FREQRANGE	Selects a frequency range for the PLL.
ENABLE	Enables the PLL.
ERROR	Error of the PLL.
AUTOTIMECONSTANT	Switches external time constant control.

Node	Short Description
TIMECONSTANT	The external time constant.
AUTOPID	Switches external PID.
PGAIN	Proportional gain of the PID controller.
ITIMECONSTANT	Integral time constant of the PID controller.
DTIMECONSTANT	Derivative Timeconstant of the PID.
FREQDELTA	Frequency deviation from center frequency.
ADCTHRESHOLD	Threshold for edge detection.
AUXAVG	Delta frequency averaging control.
SETPOINT	The setpoint in degrees of the PLL.
HARMONIC	The harmonic of the base frequency to be used.
ORDER	Selects the order of the lowpass filter.
PIDS	PID nodes.
0...n	Nodes of a PID.
INPUT	Selects the input for the PID.
INPUTCHANNEL	If applicable, selects the channel of the selected INPUT.
OUTPUT	Selects the output for the PID.
OUTPUTCHANNEL	If applicable, selects the channel of the selected OUTPUT.
OUTPUTDEFAULTENABLE	If OUTPUTDEFAULTENABLE is set, the value specified by OUTPUTDEFAULT will be applied when the PID is switched off.
OUTPUTDEFAULT	If OUTPUTDEFAULTENABLE is set, this node specifies the value to be applied.
P	Proportional gain.
I	Proportional gain for integrator.
D	Proportional gain for differentiator.
SETPOINT	Target settle point.
SETPOINTSELECT	Set point selection.
MONITOROFFSET	Offset for the monitor output.
MONITORSCALE	Scale for the monitor output.
ERROR	Shows the error value.
CENTER	Sets the output center point.
RANGE	Sets the output range.
SHIFT	Shows the output shift.

Node	Short Description
ENABLE	Enable PID controller.
RATE	Control update rate.
TIPPROTECT	Contains nodes for configuring the Tipprotect functionality.
ENABLE	Enable TipProtect for the PID controller.
PLL	Selects a PLL for TipProtect.
ACTIVE	Indicates whether TipProtect is active.
ACTIVETIMECONSTANT	Time constant when TipProtect is active.
ACTIVETHRESHOLD	Threshold for the active state.
INACTIVETIMECONSTANT	Time constant when TipProtect is inactive.
INACTIVETHRESHOLD	Threshold for the inactive state.
SIGINS	Signal Input nodes.
0...n	Nodes of a signal input.
RANGE	Voltage range for the signal input.
AC	Boolean value setting for AC coupling of the Signal Input.
IMP50	Boolean value enabling 50 Ohm input impedance termination.
DIFF	Boolean value switching differential input mode.
SIGOUTS	Signal Output nodes.
0...n	Nodes of a Signal Output.
ON	Switches the output on and off.
ADD	Switches the output adder on and off.
RANGE	Selects the output range for the Signal Output.
ENABLES	Switches for channels in the mixer.
0...n	Switches a channel in the mixer on and off.
AMPLITUDES	Amplitudes for channels in the mixer.
0...n	Fraction of the output range added to the output signal.
WAVEFORMS	
0...n	Waveforms for a channel in the mixer.
SCOPES	Scope nodes.
0...n	Nodes of a scope.
CHANNEL	Selects the channel for which scope data should be provided.

Node	Short Description
TRIGCHANNEL	Selects the channel which should be used as source for the scope's trigger.
BWLIMIT	The bandwidth-limit for the scope.
TRIGEDGE	Selects whether the scope should trigger on rising or falling edge.
TRIGLEVEL	Level at which a trigger is raised.
TRIGHOLDOFF	Time to wait for re-arming the trigger after one occurred.
TIME	Timescale of the scopewave (logarithmic decimation).
WAVE	Samples of scope-waveforms.
DIOS	DIO nodes.
0...n	Nodes of a DIO.
EXTCLK	Selects whether an external clock source should be used.
DECIMATION	Decimation for the sample rate of the DIO.
DRIVE	Selects if the outputs should be driven.
OUTPUT	Bits to output.
SYNCSELECT0	Source to output the sync signal on bit0.
SYNCSELECT1	Source to output the sync signal on bit1.
INPUT	Samples of the input.
AUXINS	Nodes of auxiliary inputs.
0...n	Node for an aux in.
AVERAGING	Averaging of the samples.
SAMPLE	Aux in samples.
AUXOUTS	Nodes of Auxiliary outputs.
0...n	Nodes of an Auxiliary output.
VALUE	Output value.
OUTPUTSELECT	Signal to be given out.
DEMODSELECT	Source demodulator.
SCALE	Scaling of the signal which is given out.
OFFSET	Value to be added to the output.
CPUS	Nodes for the real-time cpus.
0...n	Node for one real-time cpu.
WORKLOAD	Usage of the processor-time.

Node	Short Description
PROGRAM	Node to write user programs in.
OUTPUT	Node which streams output of the user-program.
USERREGS	General purpose registers to transfer data.
0...n	General purpose register.
ZCTRLS	Node containing connected ZControl devices.
CAMP0...n	A ZI current-amplifier connected to a zctrl-port.
R	Chooses a value for the shunt-resistor.
GAIN	Switches gain between 1x and 10x.
DC	Switches between AC coupling and DC coupling.
SINGLEENDED	Switches between differential and single-ended input.
TAMP0...n	A ZI transimpedance-amplifier connected to a zctrl-port.
BIASOUT	Switches between internal and external bias.
EXTBIAS	Switches the external bias.
0...n	A Channel of a transimpedance amplifier.
CURRENTGAIN	Chooses a value for the current gain.
DC	Switches between AC and DC Mode.
VOLTAGEGAIN	Chooses a value for the voltage gain.
OFFSET	Switches between internal and external bias.

7.2. Nodes

7.2.1. /ZI

This node represents the instance of the ziServer your are connected to.

7.2.2. /ZI/ABOUT

Node containing information about the server you are connected to.

7.2.3. /ZI/ABOUT/VERSION

The version of this program.

Write	-
Read	Byte Array
Setting	No

7.2.4. /ZI/ABOUT/REVISION

The revision of this program.

Write	-
Read	Integer Number
Setting	No

7.2.5. /ZI/ABOUT/FWREVISION

The revision of the used firmware.

Write	-
Read	Integer Number
Setting	No

7.2.6. /ZI/ABOUT/COPYRIGHT

The copyright string of this program.

Write	-
Read	Byte Array
Setting	No

7.2.7. /ZI/CONFIG

Configuration data of the current instance of the server.

7.2.8. /ZI/CONFIG/PORT

Configures the TCP/IP port on which the ziServer listens.

Write	Integer Number
Read	Integer Number
Setting	No
Default	8005
Range	1024 to 65535

7.2.9. /ZI/CONFIG/OPEN

Configures whether the ziServer should be open for connections from outside the local host.

Write	Integer Number	
Read	Integer Number	
Setting	No	
Unit	Boolean	
Default	0 (server only listens to localhost)	
Values	0	server only listens to localhost
	1	server is open for connections from outside

7.2.10. /ZI/TREES

Messages on tree changes.

Write	-
Read	-
Stream	Treechange message
Setting	No

Details

This node sends out a message every time a device is connected or disconnected or its tree has changed.

7.2.11. /ZI/CLOCKBASE

Provides timebase value for the server nodes

Write	-
Read	Integer Number
Setting	No

7.2.12. /DEV0...n

This node represents a single device connected to the server.

Note

The number of this node is the serial number of the connected device.

7.2.13. /DEV0...n/CLOCKBASE

Provides clockbase value for the device

Write	-
Read	Integer Number
Setting	No

7.2.14. /DEV0...n/SYSTEM

Nodes providing system information and settings.

7.2.15. /DEV0...n/SYSTEM/EXTCLK

Boolean value switching from internal to external clock.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (Internal clock)	
Values	0	Internal clock
	1	External clock

Details

When using external clock, make sure that a clock generator is connected to the Clock In connector.

7.2.16. /DEV0...n/SYSTEM/HWREVISION

The revision of the mainboard.

Write	-
Read	Integer Number
Setting	No

7.2.17. /DEV0...n/FEATURES

Node containing information on features of the device.

7.2.18. /DEV0...n/FEATURES/SERIAL

Node providing the serial number of the device.

Write	-
Read	Byte Array
Setting	No

7.2.19. /DEV0...n/FEATURES/DEVTYPE

Node providing a string about the type of device.

Write	-
Read	Byte Array
Setting	No

7.2.20. /DEV0...n/FEATURES/OPTIONS

Node giving information on enabled options.

Write	-
Read	Byte Array
Setting	No

Details

Reading this node returns a string containing a newline-separated list of all installed options.

7.2.21. /DEV0...n/FEATURES/CODE

Node providing a mechanism to write feature codes.

Write	Byte Array
Read	-
Setting	No

7.2.22. /DEV0...n/STATUS

Nodes providing status information from the device.

7.2.23. /DEV0...n/STATUS/TIME

The current timestamp.

Write	-
Read	Decimal Number
Setting	No
Unit	s

7.2.24. /DEV0...n/STATUS/FLAGS

Node containing some status flags.

7.2.25. /DEV0...n/STATUS/FLAGS/BINARY

A binary representation of all flags.

Write	-	
Read	Integer Number	
Setting	No	
Unit	bit-coded	
Values	b0 = 1	PLL unlocked
	b1 = 2	HF clock unlocked
	b2 = 4	FX2 RX error
	b3 = 8	Package loss
	b4 = 16	Output 1 clipped
	b5 = 32	Output 2 clipped
	b6 = 64	Input 1 clipped
	b7 = 128	Input 2 clipped
	b8 = 256	Scope skipped a shot
	b9 = 512	FX2 TX buffer almost full
	b10 = 1024	0
	b11 = 2048	PLL unlocked (version without debouncing)
	b12 = 4096	FX2 TX package lost

Details

When multiple flags are set the values are or-ed.

7.2.26. /DEV0...n/STATUS/FLAGS/PLLLOCK

Flag indicating if the internal PLL for clock generation has locked.

Write	-	
Read	Integer Number	
Setting	No	
Unit	Boolean	
Values	0	PLL locked
	1	PLL not locked

7.2.27. /DEV0...n/STATUS/FLAGS/DCMLOCK

Flag indicating if the internal digital clock manager (DCM) has locked.

Write	-	
Read	Integer Number	
Setting	No	
Unit	Boolean	
Values	0	DCM locked
	1	DCM not locked

7.2.28. /DEV0...n/STATUS/FLAGS/FX2RX

Flag indicating if the device receives data via USB.

Write	-	
Read	Integer Number	
Setting	No	
Unit	Boolean	
Values	0	Device receives
	1	Device does not receive

7.2.29. /DEV0...n/STATUS/FLAGS/PKGLOSS

Flag indicating that the device lost data when sending via USB.

Write	-

Read	Integer Number	
Setting	No	
Unit	Boolean	
Values	0	no packetloss
	1	packets are lost

7.2.30. /DEV0...n/STATUS/FLAGS/MIXERCLIP

Flags indicating that the internal mixer is clipping.

7.2.31. /DEV0...n/STATUS/FLAGS/MIXERCLIP/0...n

Flag indicating that this mixer-channel is clipping.

Write	-	
Read	Integer Number	
Setting	No	
Unit	Boolean	
Values	0	no clipping
	1	clipping

7.2.32. /DEV0...n/STATUS/FLAGS/ADCCLIP

Flags indicating that the AD-converter is clipping.

7.2.33. /DEV0...n/STATUS/FLAGS/ADCCLIP/0...n

Flag indicating that this ADC-channel is clipping.

Write	-
Read	Integer Number
Setting	No
Unit	Boolean

Values	0	no clipping
	1	clipping

7.2.34. /DEV0...n/STATUS/FLAGS/SCOPESKIPPED

Flag indicating that scope data has been skipped.

Write	-	
Read	Integer Number	
Setting	No	
Unit	Boolean	
Values	0	no data skipped
	1	data skipped

Details

This happens when too much data is being sent over USB.

7.2.35. /DEV0...n/STATUS/FLAGS/DEMODSAMPLELOSS

Flag indicating that demod data has been lost.

Write	-	
Read	Integer Number	
Setting	No	
Unit	Boolean	
Values	0	no demod data lost
	1	demod data lost

7.2.36. /DEV0...n/STATUS/FIFOLEVEL

Percentage of tx-fifo used.

Write	-
Read	Decimal Number
Setting	No
Unit	Percent

Range	0 to 100

7.2.37. /DEV0...n/STATUS/ADC0MIN

The minimum value on Signal Input 1 (ADC0) during 100 ms.

Write	-
Read	Integer Number
Setting	No
Range	-127 to 127

7.2.38. /DEV0...n/STATUS/ADC0MAX

The maximum value on Signal Input 1 (ADC0) during 100 ms.

Write	-
Read	Integer Number
Setting	No
Range	-127 to 127

7.2.39. /DEV0...n/STATUS/ADC1MIN

The minimum value on Signal Input 2 (ADC1) during 100 ms.

Write	-
Read	Integer Number
Setting	No
Range	-127 to 127

7.2.40. /DEV0...n/STATUS/ADC1MAX

The maximum value on Signal Input 2 (ADC1) during 100 ms.

Write	-

Read	Integer Number
Setting	No
Range	-127 to 127

7.2.41. /DEV0...n/STATS

Nodes providing statistical data about the device.

7.2.42. /DEV0...n/STATS/BYTESSENT

Total amount of bytes sent via USB.

Write	-
Read	Integer Number
Setting	No
Unit	Bytes

7.2.43. /DEV0...n/STATS/BYTESRECEIVED

Total amount of bytes received via USB.

Write	-
Read	Integer Number
Setting	No
Unit	Bytes

7.2.44. /DEV0...n/STATS/MEANPOLLCNT

Average poll-count.

Write	-
Read	Decimal Number
Setting	No
Unit	Polls/Second

7.2.45. /DEV0...n/STATS/MEANMSGCNT

Average message-count.

Write	-
Read	Decimal Number
Setting	No
Unit	Messages/Second

7.2.46. /DEV0...n/STATS/PHYSICAL

Group of nodes providing some physical information on the device.

7.2.47. /DEV0...n/STATS/PHYSICAL/1V2

Actual voltage of the 1.2volts supply.

Write	-
Read	Decimal Number
Setting	No
Unit	Volts

7.2.48. /DEV0...n/STATS/PHYSICAL/1V8

Actual voltage of the 1.8volts supply.

Write	-
Read	Decimal Number
Setting	No
Unit	Volts

7.2.49. /DEV0...n/STATS/PHYSICAL/2V5

Actual voltage of the 2.5volts supply.

Write	-
Read	Decimal Number
Setting	No
Unit	Volts

7.2.50. /DEV0...n/STATS/PHYSICAL/3V3

Actual voltage of the 3.3volts supply.

Write	-
Read	Decimal Number
Setting	No
Unit	Volts

7.2.51. /DEV0...n/STATS/PHYSICAL/5V0

Actual voltage of the 5.0volts supply.

Write	-
Read	Decimal Number
Setting	No
Unit	Volts

7.2.52. /DEV0...n/STATS/PHYSICAL/TEMP

Actual temperature.

Write	-
Read	Decimal Number
Setting	No
Unit	Degrees Celsius

7.2.53. /DEV0...n/DEMODS

Demodulator nodes.

7.2.54. /DEV0...n/DEMODS/0...n

Nodes of a single demodulator

7.2.55. /DEV0...n/DEMODS/0...n/ADCSELECT

Selects the index of the signal input for the demodulator.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Index	
Range	0 to 5	
Values	0	Signal input 0
	1	Signal input 1
	2	Aux Input 0
	3	Aux Input 1
	4	DIO 0
	5	DIO 1

7.2.56. /DEV0...n/DEMODS/0...n/ORDER

Selects the order of the low-pass filter.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Order	
Default	4 (24dB/oct slope)	
Range	1 to 8	
Values	1	6dB/oct slope
	2	12dB/oct slope
	3	18dB/oct slope
	4	24dB/oct slope

	5	30dB/oct slope
	6	36dB/oct slope
	7	42dB/oct slope
	8	48dB/oct slope

7.2.57. /DEV0...n/DEMODS/0...n/TIMECONSTANT

Sets the time constant of the low-pass filter.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	s
Default	0.010164

Details

The time constant is set for each stage of the low-pass filter. The total timeconstant and bandwidth depends on the selected order.

7.2.58. /DEV0...n/DEMODS/0...n/RATE

The number of output values sent to the computer per second.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	Hz

7.2.59. /DEV0...n/DEMODS/0...n/TRIGGER

Sets the trigger- and gating-functionality of the demodulator.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	bit-coded	
Values	b0 = 1	DIO0 rising edge

	b1 = 2	DIO0 falling edge
	b2 = 4	DIO1 rising edge
	b3 = 8	DIO1 falling edge
	b4 = 16	DIO0 high
	b5 = 32	DIO0 low
	b6 = 64	DIO1 high
	b7 = 128	DIO1 low

Details

The triggers are configured by the bits of an integer. When multiple bits/triggers are set, they are or-ed. If trigger is set to 0 then demodulator data is sent continuously.

7.2.60. /DEV0...n/DEMODS/0...n/OSCSELECT

Index of the oscillator used to demodulate the signal.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Index

7.2.61. /DEV0...n/DEMODS/0...n/HARMONIC

The harmonic of the base frequency to be used.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Harmonic
Default	1
Range	1 to 1023

Details

Selecting 1 chooses the fundamental frequency.

7.2.62. /DEV0...n/DEMODS/0...n/PHASESHIFT

Phaseshift of the demodulator.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	deg
Default	0
Range	-180 to 180

7.2.63. /DEV0...n/DEMODS/0...n/SINC

Boolean value enabling Sinc filter functionality.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (Sinc filtering disabled)	
Values	0	Sinc filtering disabled
	1	Sinc filtering enabled

7.2.64. /DEV0...n/DEMODS/0...n/SAMPLE

Samples of the demodulator are given out at this node.

Write	-
Read	Demodulator Sample
Stream	Demodulator Sample
Setting	No

7.2.65. /DEV0...n/OSCS

Oscillator nodes.

7.2.66. /DEV0...n/OSCS/0...n

Nodes of an oscillator.

7.2.67. /DEV0...n/OSCS/0...n/FREQ

Frequency to of the oscillator.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	Hz
Default	1.0E6
Range	0 to 1.0E8

7.2.68. /DEV0...n/MODS

Modulator option nodes.

Details

These nodes are only visible when the MOD option is installed on the device.

7.2.69. /DEV0...n/MODS/0...n

Nodes of a Modulator.

Note

These nodes are only available with installed Modulation-Option.

7.2.70. /DEV0...n/MODS/0...n/FMCOEFF

Frequency Modulation Coefficient.

Note

this node is only active in FM-mode (MODE=3).

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Default	0
Range	0 to 12.56618

7.2.71. /DEV0...n/MODS/0...n/MODE

Modulation mode.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Default	0 (Off)	
Values	0	Off
	1	Amplitude modulation
	2	Frequency demodulation
	3	Frequency modulation

7.2.72. /DEV0...n/MODS/0...n/SIDEBAND

Sideband selector.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Default	0 (Both)	
Values	0	Both
	1	Upper
	2	Lower

7.2.73. /DEV0...n/MODS/0...n/SAMPLE

Modulation Samples.

Write	-
Read	Demodulator Sample

Stream	Demodulator Sample
Setting	No

7.2.74. /DEV0...n/PLLS

PLL nodes.

7.2.75. /DEV0...n/PLLS/0...n

Nodes of a PLL.

Note

These nodes are only available with installed PLL option.

7.2.76. /DEV0...n/PLLS/0...n/ADCSELECT

Selects an input for the PLL.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Index	
Range	0 to 5	
Values	0	Signal Input 1
	1	Signal Input 2
	2	Aux Input 1
	3	Aux Input 2
	4	DIO 0
	5	DIO 1

7.2.77. /DEV0...n/PLLS/0...n/AUTOCENTER

Switches autocenter.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Boolean
Default	0 (Autocenter off)
Values	0 Autocenter off 1 Autocenter on

7.2.78. /DEV0...n/PLLS/0...n/FREQCENTER

Selects a center frequency.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	Hz
Range	0 to 1.0E8

7.2.79. /DEV0...n/PLLS/0...n/FREQRANGE

Selects a frequency range for the PLL.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	Hz
Range	0 to 1.0E8

7.2.80. /DEV0...n/PLLS/0...n/ENABLE

Enables the PLL.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Boolean

Default	0 (PLL off)	
Values	0	PLL off
	1	PLL on

7.2.81. /DEV0...n/PLLS/0...n/ERROR

Error of the PLL.

Write	-
Read	Decimal Number
Stream	Decimal Number
Setting	No
Unit	deg
Range	-180 to 180

7.2.82. /DEV0...n/PLLS/0...n/AUTOTIMECONSTANT

Switches external time constant control.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (External time constant off)	
Values	0	External time constant off
	1	External time constant on

7.2.83. /DEV0...n/PLLS/0...n/TIMECONSTANT

The external time constant.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	s
Default	0

7.2.84. /DEV0...n/PLLS/0...n/AUTOPID

Switches external PID.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Boolean
Default	0 (External PID off)
Values	0 External PID off 1 External PID on

7.2.85. /DEV0...n/PLLS/0...n/PGAIN

Proportional gain of the PID controller.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	Hz/deg
Default	0
Range	-72904.8 to 72904.24

7.2.86. /DEV0...n/PLLS/0...n/ITIMECONSTANT

Integral time constant of the PID controller.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	Hz^2/deg
Default	145.48991

7.2.87. /DEV0...n/PLLS/0...n/DTIMECONSTANT

Derivative Timeconstant of the PID.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	1/deg
Default	0

7.2.88. /DEV0...n/PLLS/0...n/FREQDELTA

Frequency deviation from center frequency.

Write	-
Read	Decimal Number
Setting	No
Unit	Hz

7.2.89. /DEV0...n/PLLS/0...n/ADCTHRESHOLD

Threshold for edge detection.

Write	Integer Number
Read	Integer Number
Setting	Yes
Default	100
Range	-4096 to 4095

Details

Full scale corresponds to -4096 and 4095.

7.2.90. /DEV0...n/PLLS/0...n/AUXAVG

Delta frequency averaging control.

Write	Integer Number
Read	Integer Number
Setting	Yes
Default	8

Range	0 to 128

7.2.91. /DEV0...n/PLLS/0...n/SETPOINT

The setpoint in degrees of the PLL.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	deg
Default	0
Range	-180 to 180

7.2.92. /DEV0...n/PLLS/0...n/HARMONIC

The harmonic of the base frequency to be used.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Harmonic
Default	1
Range	1 to 1023

Details

Selecting 1 chooses the fundamental frequency.

7.2.93. /DEV0...n/PLLS/0...n/ORDER

Selects the order of the lowpass filter.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Order
Default	4 (24dB/oct slope)
Range	1 to 8

Values	1	6dB/oct slope
	2	12dB/oct slope
	3	18dB/oct slope
	4	24dB/oct slope
	5	30dB/oct slope
	6	36dB/oct slope
	7	42dB/oct slope
	8	48dB/oct slope

7.2.94. /DEV0...n/PIDS

PID nodes.

7.2.95. /DEV0...n/PIDS/0...n

Nodes of a PID.

Note

These nodes are only available with installed PID option.

7.2.96. /DEV0...n/PIDS/0...n/INPUT

Selects the input for the PID.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Default	0 (Demodulator X value [Vrms])	
Values	0	Demodulator X value [Vrms]
	1	Demodulator Y value [Vrms]
	2	Demodulator R value [Vrms]
	3	Demodulator Theta value [deg]
	4	Auxin [V]
	5	Auxout (as input) [V]
	6	Modulation Index [0,1]

	7	Dual Frequency Tracking |Z(n+)| - |Z(n)| [Vrms]
	8	Demodulator x(n+1) - x(n) [Vrms]
	9	Demodulator |z(n+1) - z(n)| [Vrms]
	10	Oscillator Frequency [Hz]

7.2.97. /DEV0...n/PIDS/0...n/INPUTCHANNEL

If applicable, selects the channel of the selected INPUT.

Write	Integer Number
Read	Integer Number
Setting	Yes
Default	0

Details

Sets the input channel index for the selected INPUT, i.e., 0,1,2 etc.. The available channels depend on the input type.

7.2.98. /DEV0...n/PIDS/0...n/OUTPUT

Selects the output for the PID.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Default	0 (Sigout 1 [Vrms])	
Values	0	Sigout 1 [Vrms]
	1	Sigout 2 [Vrms]
	2	Oscout [Hz]
	3	Auxout (manual mode) [V]
	4	DIO [5V-TTL]

7.2.99. /DEV0...n/PIDS/0...n/OUTPUTCHANNEL

If applicable, selects the channel of the selected OUTPUT.

Write	Integer Number
Read	Integer Number
Setting	Yes
Default	6

Details

Sets the input channel index for the selected OUTPUT, i.e., 0,1,2 etc.. The available channels depend on the output type

7.2.100. /DEV0...n/PIDS/0...n/OUTPUTDEFAULTENABLE

If OUTPUTDEFAULTENABLE is set, the value specified by OUTPUTDEFAULT will be applied when the PID is switched off.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	No	
Values	0	OFF
	1	ON

Details

7.2.101. /DEV0...n/PIDS/0...n/OUTPUTDEFAULT

If OUTPUTDEFAULTENABLE is set, this node specifies the value to be applied.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Default	0.1

Details

7.2.102. /DEV0...n/PIDS/0...n/P

Proportional gain.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	[OUTPUT Unit]/[INPUT Unit]
Default	1

Details

Sets the proportional gain for the error signal. Negative feedback corresponds to a negative gain.

7.2.103. /DEV0...n/PIDS/0...n/I

Proportional gain for integrator.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	[OUTPUT Unit]/[INPUT Unit]/s
Default	10

Details

Sets the proportional gain for the integrated (accumulated) error signal. Negative feedback corresponds to a negative gain.

7.2.104. /DEV0...n/PIDS/0...n/D

Proportional gain for differentiator.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	[OUTPUT Unit]/[INPUT Unit]*s
Default	0.001

Details

Sets the proportional gain for the differentiated error signal. Negative feedback corresponds to a negative gain.

7.2.105. /DEV0...n/PIDS/0...n/SETPOINT

Target settle point.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	[INPUT Unit]
Default	0.1

7.2.106. /DEV0...n/PIDS/0...n/SETPOINTSELECT

Set point selection.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Default	0 (Manual Setpoint)	
Values	0	Manual Setpoint
	1	Aux Input 1
	2	Aux Input 2
	3	PID n

7.2.107. /DEV0...n/PIDS/0...n/MONITOROFFSET

Offset for the monitor output.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Default	0

7.2.108. /DEV0...n/PIDS/0...n/MONITORSCALE

Scale for the monitor output.

Write	Decimal Number
Read	Decimal Number

Setting	Yes
Default	1

7.2.109. /DEV0...n/PIDS/0...n/ERROR

Shows the error value.

Write	-
Read	Decimal Number
Setting	No
Unit	[OUTPUT Unit]

Details

The calculated error is : ERROR = SETPOINT - IN.

7.2.110. /DEV0...n/PIDS/0...n/CENTER

Sets the output center point.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	[OUTPUT Unit]
Default	0.1

7.2.111. /DEV0...n/PIDS/0...n/RANGE

Sets the output range.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	[OUTPUT Unit]
Default	0.5

Details

The limits for the output are : OUT = [CENTER - RANGE,CENTER + RANGE] with RANGE > 0.0.

7.2.112. /DEV0...n/PIDS/0...n/SHIFT

Shows the output shift.

Write	-
Read	Decimal Number
Setting	No
Unit	[OUTPUT Unit]
Default	0

Details

The calculated output value is : OUT = CENTER + SHIFT.

7.2.113. /DEV0...n/PIDS/0...n/ENABLE

Enable PID controller.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (OFF)	
Values	0	OFF
	1	ON

7.2.114. /DEV0...n/PIDS/0...n/RATE

Control update rate.

Write	-
Read	Decimal Number
Setting	No
Unit	Samples/s

7.2.115. /DEV0...n/PIDS/0...n/TIPPROTECT

Contains nodes for configuring the Tipprotect functionality.

7.2.116. /DEV0...n/PIDS/0...n/TIPPROTECT/ENABLE

Enable TipProtect for the PID controller.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Boolean
Default	0 (OFF)
Values	0 OFF 1 ON

7.2.117. /DEV0...n/PIDS/0...n/TIPPROTECT/PLL

Selects a PLL for TipProtect.

Write	Integer Number
Read	Integer Number
Setting	Yes
Default	0

7.2.118. /DEV0...n/PIDS/0...n/TIPPROTECT/ACTIVE

Indicates whether TipProtect is active.

Write	-
Read	Integer Number
Setting	No
Unit	Boolean
Default	0

7.2.119. /DEV0...n/PIDS/0...n/TIPPROTECT/ACTIVETIMECONSTANT

Time constant when TipProtect is active.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	s
Default	0
Range	0 to 1.0E8

Details

Time constant for low-pass filtering the PLL error² when TipProtect is active, i.e., when waiting to reenable the PID controller.

7.2.120. /DEV0...n/PIDS/0...n/TIPPROTECT/ ACTIVETHRESHOLD

Threshold for the active state.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	deg²
Range	0 to 129600

Details

Threshold for PLL error² when TipProtect is active, i.e., when waiting to reenable the PID controller.

7.2.121. /DEV0...n/PIDS/0...n/TIPPROTECT/ INACTIVETIMECONSTANT

Time constant when TipProtect is inactive.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	s
Default	0
Range	0 to 1.0E8

Details

Time constant for low-pass filtering the PLL error² when TipProtect is inactive, i.e., when waiting to disable the PID controller.

7.2.122. /DEV0...n/PIDS/0...n/TIPPROTECT/INACTIVETHRESHOLD

Threshold for the inactive state.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	deg²
Range	0 to 129600

Details

Threshold for PLL error² when TipProtect is inactive, i.e., when waiting to disable the PID controller.

7.2.123. /DEV0...n/SIGINS

Signal Input nodes.

7.2.124. /DEV0...n/SIGINS/0...n

Nodes of a signal input.

7.2.125. /DEV0...n/SIGINS/0...n/RANGE

Voltage range for the signal input.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	V
Default	1.2
Range	0.0001 to 2

7.2.126. /DEV0...n/SIGINS/0...n/AC

Boolean value setting for AC coupling of the Signal Input.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (DC coupling)	
Values	0	DC coupling
	1	AC coupling

7.2.127. /DEV0...n/SIGINS/0...n/IMP50

Boolean value enabling 50 Ohm input impedance termination.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (High impedance)	
Values	0	High impedance
	1	50 Ohm impedance

7.2.128. /DEV0...n/SIGINS/0...n/DIFF

Boolean value switching differential input mode.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (Single-ended inputs)	
Values	0	Single-ended inputs
	1	Differential inputs

7.2.129. /DEV0...n/SIGOUTS

Signal Output nodes.

7.2.130. /DEV0...n/SIGOUTS/0...n

Nodes of a Signal Output.

7.2.131. /DEV0...n/SIGOUTS/0...n/ON

Switches the output on and off.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (Output off)	
Values	0	Output off
	1	Output on

7.2.132. /DEV0...n/SIGOUTS/0...n/ADD

Switches the output adder on and off.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (Adder off)	
Values	0	Adder off
	1	Adder on

7.2.133. /DEV0...n/SIGOUTS/0...n/RANGE

Selects the output range for the Signal Output.

Write	Decimal Number

Read	Decimal Number	
Setting	Yes	
Unit	V	
Default	1	
Values	0.01	0.01 V range
	0.1	0.1 V range
	1	1 V range
	10	10 V range

7.2.134. /DEV0...n/SIGOUTS/0...n/ENABLES

Switches for channels in the mixer.

7.2.135. /DEV0...n/SIGOUTS/0...n/ENABLES/0...n

Switches a channel in the mixer on and off.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Values	0	Channel off (unconditionally)
	1	Channel on (unconditionally)
	2	Channel off (will be turned off on next change of sign from negative to positive)
	3	Channel on (will be turned on on next change of sign from negative to positive)

7.2.136. /DEV0...n/SIGOUTS/0...n/AMPLITUDES

Amplitudes for channels in the mixer.

7.2.137. /DEV0...n/SIGOUTS/0...n/AMPLITUDES/0...n

Fraction of the output range added to the output signal.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	Gain
Range	-1 to 1

Details

Multiply this value with the range setting to obtain voltage in V.

7.2.138. /DEV0...n/SIGOUTS/0...n/WAVEFORMS

7.2.139. /DEV0...n/SIGOUTS/0...n/WAVEFORMS/0...n

Waveforms for a channel in the mixer.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Default	0 (Sine)	
Values	0	Sine
	1	Square

Details

For hardware revisions 1.4 and lower, the output signal range for rectangular output is limited to 1V.

7.2.140. /DEV0...n/SCOPES

Scope nodes.

7.2.141. /DEV0...n/SCOPES/0...n

Nodes of a scope.

7.2.142. /DEV0...n/SCOPES/0...n/CHANNEL

Selects the channel for which scope data should be provided.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Index	
Default	0 (Signal Input 1)	
Range	0 to 3	
Values	0	Signal Input 1
	1	Signal Input 2
	2	Signal Output 1
	3	Signal Output 2

7.2.143. /DEV0...n/SCOPES/0...n/TRIGCHANNEL

Selects the channel which should be used as source for the scope's trigger.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Index	
Default	-1 (Off)	
Range	-2 to 11	
Values	-2	Continuous
	-1	Off
	0	Signal Input 1
	1	Signal Input 2
	2	Signal Output 1
	3	Signal Output 2
	4	Ocillator 1 phase
	5	Ocillator 2 phase
	6	Ocillator 3 phase
	7	Ocillator 4 phase
	8	Ocillator 5 phase
	9	Ocillator 6 phase

	0	Ocillator 7 phase
	11	Ocillator 8 phase
	12	DIO 0
	13	DIO 1

7.2.144. /DEV0...n/SCOPES/0...n/BWLIMIT

The bandwidth-limit for the scope.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (BW-limit off)	
Values	0	BW-limit off
	1	BW-limit on

7.2.145. /DEV0...n/SCOPES/0...n/TRIGEDGE

Selects whether the scope should trigger on rising or falling edge.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Default	1 (Rising edge)	
Values	0	Falling edge
	1	Rising edge

7.2.146. /DEV0...n/SCOPES/0...n/TRIGLEVEL

Level at which a trigger is raised.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	LSB

Default	0
Range	-32768 to 32767

Details

Full scale is covered by min and max values

7.2.147. /DEV0...n/SCOPES/0...n/TRIGHOLDOFF

Time to wait for re-arming the trigger after one occurred.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	s
Default	0.25
Range	0 to 65.535

7.2.148. /DEV0...n/SCOPES/0...n/TIME

Timescale of the scopewave (logarithmic decimation).

Write	Integer Number
Read	Integer Number
Setting	Yes
Default	0
Range	0 to 7

Details

Determines the decimation of the sample rate. The following formulas apply: span = 2^val * 10us, samplerate = 210MS/2^val

7.2.149. /DEV0...n/SCOPES/0...n/WAVE

Samples of scope-waveforms.

Write	-
Read	Scope Sample
Stream	Scope Sample
Setting	No

7.2.150. /DEV0...n/DIOS

DIO nodes.

7.2.151. /DEV0...n/DIOS/0...n

Nodes of a DIO.

7.2.152. /DEV0...n/DIOS/0...n/EXTCLK

Selects whether an external clock source should be used.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (Internal clock)	
Values	0	Internal clock
	1	External clock

Details

The external clock needs to be applied to the DIO connector when this node is set to 1.

7.2.153. /DEV0...n/DIOS/0...n/DECIMATION

Decimation for the sample rate of the DIO.

Write	Integer Number
Read	Integer Number
Setting	Yes
Default	256
Range	0 to 65535

7.2.154. /DEV0...n/DIOS/0...n/DRIVE

Selects if the outputs should be driven.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Default	0 (Drive off)	
Range	0 to 3	
Values	0	Drive off
	1	Drive lower 8bits
	2	Drive higher 8bits
	3	Drive all 16bits

7.2.155. /DEV0...n/DIOS/0...n/OUTPUT

Bits to output.

Write	Integer Number
Read	Integer Number
Setting	Yes
Default	0
Range	0 to 65535

7.2.156. /DEV0...n/DIOS/0...n/SYNCSELECT0

Source to output the sync signal on bit0.

Write	Integer Number
Read	Integer Number
Setting	Yes
Default	8
Range	0 to 8

7.2.157. /DEV0...n/DIOS/0...n/SYNCSELECT1

Source to output the sync signal on bit1.

Write	Integer Number
Read	Integer Number

Setting	Yes
Default	8
Range	0 to 8

7.2.158. /DEV0...n/DIOS/0...n/INPUT

Samples of the input.

Write	-
Read	DIO Sample
Stream	DIO Sample
Setting	No

7.2.159. /DEV0...n/AUXINS

Nodes of auxiliary inputs.

7.2.160. /DEV0...n/AUXINS/0...n

Node for an aux in.

7.2.161. /DEV0...n/AUXINS/0...n/AVERAGING

Averaging of the samples.

Write	Integer Number
Read	Integer Number
Setting	Yes
Default	256
Range	1 to 32768

7.2.162. /DEV0...n/AUXINS/0...n/SAMPLE

Aux in samples.

Write	-
Read	Aux input Sample
Stream	Aux input Sample
Setting	No

7.2.163. /DEV0...n/AUXOUTS

Nodes of Auxiliary outputs.

7.2.164. /DEV0...n/AUXOUTS/0...n

Nodes of an Auxiliary output.

7.2.165. /DEV0...n/AUXOUTS/0...n/VALUE

Output value.

Write	-
Read	Decimal Number
Setting	No
Unit	V
Range	-10 to 10

7.2.166. /DEV0...n/AUXOUTS/0...n/OUTPUTSELECT

Signal to be given out.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Values	-1	Manual
	0	X

	1	Y
	2	R
	3	Theta
	4	PLL 0 (with installed PLL option))
	4	PLL 1 (with installed PLL option))

7.2.167. /DEV0...n/AUXOUTS/0...n/DEMODSELECT

Source demodulator.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Index
Default	0

7.2.168. /DEV0...n/AUXOUTS/0...n/SCALE

Scaling of the signal which is given out.

Write	Decimal Number
Read	Decimal Number
Setting	Yes

7.2.169. /DEV0...n/AUXOUTS/0...n/OFFSET

Value to be added to the output.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	V
Default	0
Range	-2560 to 2560

Details

The offset value is applied after scaling.

7.2.170. /DEV0...n/CPUS

Nodes for the real-time cpus.

7.2.171. /DEV0...n/CPUS/0...n

Node for one real-time cpu.

Note

Only available with installed real-time option.

7.2.172. /DEV0...n/CPUS/0...n/WORKLOAD

Usage of the processor-time.

Write	-
Read	Decimal Number
Setting	No
Range	0 to 1

7.2.173. /DEV0...n/CPUS/0...n/PROGRAM

Node to write user programs in.

Write	Byte Array
Read	-
Setting	No

7.2.174. /DEV0...n/CPUS/0...n/OUTPUT

Node which streams output of the user-program.

Write	-
Read	-
Stream	Byte Array

Setting	No

7.2.175. /DEV0...n/CPUS/0...n/USERREGS

General purpose registers to transfer data.

7.2.176. /DEV0...n/CPUS/0...n/USERREGS/0...n

General purpose register.

Write	Integer Number
Read	Integer Number
Setting	Yes

7.2.177. /DEV0...n/ZCTRLS

Node containing connected ZControl devices.

Note

There has to be a device connected to either ZCtrl1 or ZCtrl2 on the backplane that children of this node appear.

7.2.178. /DEV0...n/ZCTRLS/CAMP0...n

A ZI current-amplifier connected to a zctrl-port.

Note

An HF2-CAmp has to be connected to the device in order for this node to appear.

7.2.179. /DEV0...n/ZCTRLS/CAMP0...n/R

Chooses a value for the shunt-resistor.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Ohm
Default	0 (open, high ohmic)
Values	0 open, high ohmic 10 10 Ohm 100 100 Ohm 1000 1 kOhm 10000 10 kOhm 100000 100 kOhm 1.0E6 1 MOhm

7.2.180. /DEV0...n/ZCTRLS/CAMP0...n/GAIN

Switches gain between 1x and 10x.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Gain
Default	1 (1x gain)
Values	1 1x gain 10 10x gain

7.2.181. /DEV0...n/ZCTRLS/CAMP0...n/DC

Switches between AC coupling and DC coupling.

Write	Integer Number
Read	Integer Number
Setting	Yes
Unit	Boolean
Default	0 (AC coupling)
Values	0 AC coupling 1 DC coupling

7.2.182. /DEV0...n/ZCTRLS/CAMP0...n/SINGLEENDED

Switches between differential and single-ended input.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	0 (Differential inputs)	
Values	0	Differential inputs
	1	Single-ended inputs

7.2.183. /DEV0...n/ZCTRLS/TAMP0...n

A ZI transimpedance-amplifier connected to a zctrl-port.

Note

An HF2-TAmp has to be connected to the device in order that this node appears.

7.2.184. /DEV0...n/ZCTRLS/TAMP0...n/BIASOUT

Switches between internal and external bias.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	V
Default	0

7.2.185. /DEV0...n/ZCTRLS/TAMP0...n/EXTBIAS

Switches the external bias.

Write	Integer Number
Read	Integer Number
Setting	Yes

Unit	Boolean	
Default	0 (External bias off)	
Values	0	External bias off
	1	External bias on

7.2.186. /DEV0...n/ZCTRLS/TAMP0...n/0...n

A Channel of a transimpedance amplifier.

7.2.187. /DEV0...n/ZCTRLS/TAMP0...n/0...n/CURRENTGAIN

Chooses a value for the current gain.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Gain	
Default	1000 (1kx)	
Values	100	100x
	1000	1kx
	10000	10kx
	100000	100kx
	1.0E6	1Mx
	1.0E7	10Mx
	1.0E8	100Mx

7.2.188. /DEV0...n/ZCTRLS/TAMP0...n/0...n/DC

Switches between AC and DC Mode.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Boolean	
Default	1 (DC Mode)	
Values	0	AC Mode

	1	DC Mode

7.2.189. /DEV0...n/ZCTRLS/TAMP0...n/0...n/VOLTAGEGAIN

Chooses a value for the voltage gain.

Write	Integer Number	
Read	Integer Number	
Setting	Yes	
Unit	Gain	
Default	1 (1x)	
Values	1	1x
	10	10x

7.2.190. /DEV0...n/ZCTRLS/TAMP0...n/0...n/OFFSET

Switches between internal and external bias.

Write	Decimal Number
Read	Decimal Number
Setting	Yes
Unit	V
Default	0

Chapter 8. Real-time Option

The Real-time option provides the capability to execute programs written in the C programming language on the RISC microprocessor of the HF2 Instrument with predictable latency and comes with an extensive programming environment.

This chapter describes:

- Installation of the Real-time programming environment, in Section 8.1 . See Section 8.1.1 and Section 8.1.2 for the installation process on Windows and Linux, respectively. Section 8.1.3 explains where to find the documentation in HTML format.
- The Real-time Option programming reference guide which provides examples and lists the data structures and functions available in the API, in Section 8.2 .

Note

RT programming can be used only if the HF2LI-RT / HF2IS-RT option has been purchased and activated. Customers can purchase the RT option at any time, whether when ordering their instrument or after delivery. This option can be activated by the user or by Zurich Instruments via remote servicing.

Note

The Real-time Option programming reference guide is also available as HTML. The HTML documentation is bundled with the Real-time installation zip-file available from Zurich Instruments' download page [http://www.zhinst.com/downloads].

8.1. Installation of the Real-time Development Environment

In this section we describe the installation process of the HF2's real-time development environment, see Section 8.1.1 for Windows and Section 8.1.2 for Linux. The real-time development environment is available from Zurich Instruments' download page [http://www.zhinst.com/downloads].

Note

The RT development environment does not include a special editor. Please use an editor of your choice, for example:

- notepad++ [http://notepad-plus-plus.org] or PSPad [http://www.pspad.com] on Windows,
- emacs, vim, etc. on Linux.

8.1.1. Installation on Windows

Software Requirements

To use the compilation tools on Windows the RT development environment requires the 32-bit version of Cygwin which provides a Linux-like environment. Cygwin is free and open source software, for more details see the Cygwin website [http://www.cygwin.com]. The only Cygwin package necessary is the `make` package. Installation of Cygwin is also detailed below.

Note

Even if your PC is natively 64-bit, the 32-bit version of Cygwin is required to run the compiler tools distributed with the RT development environment.

Installation Steps

1. Download the 32-bit version of Cygwin [http://cygwin.com/setup-x86.exe] and run the Setup executable.

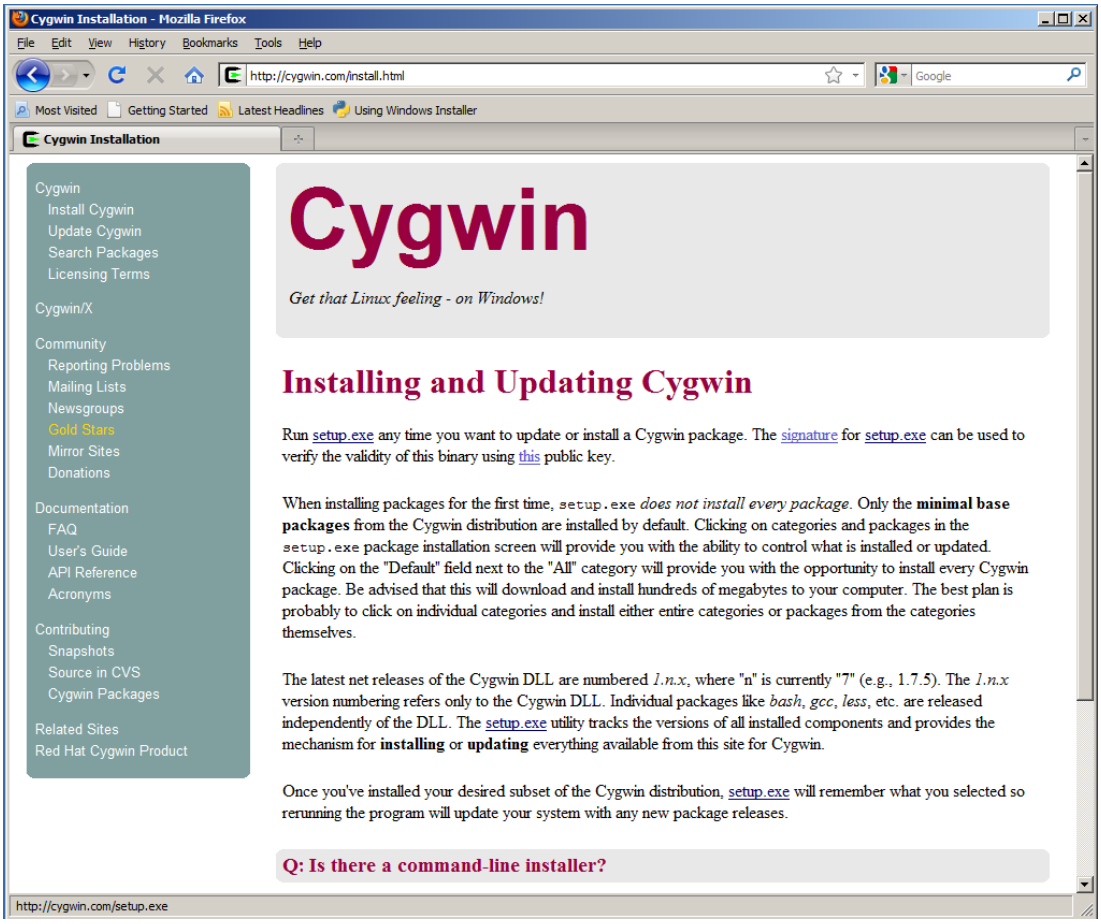

Cygwin Installation - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://cygwin.com/install.html
Most Visited Getting Started Latest Headlines Using Windows Installer
Cygwin Installation
Cygwin
Install Cygwin
Update Cygwin
Search Packages
Licensing Terms
Cygwin/X
Community
Reporting Problems
Mailing Lists
Newsgroups
Gold Stars
Mirror Sites
Donations
Documentation
FAQ
User's Guide
API Reference
Acronyms
Contributing
Snapshots
Source in CVS
Cygwin Packages
Related Sites
Red Hat Cygwin Product
Cygwin
Get that Linux feeling - on Windows!
Installing and Updating Cygwin
Run setup.exe any time you want to update or install a Cygwin package. The signature for setup.exe can be used to verify the validity of this binary using this public key.
When installing packages for the first time, setup.exe does not install every package. Only the minimal base packages from the Cygwin distribution are installed by default. Clicking on categories and packages in the setup.exe package installation screen will provide you with the ability to control what is installed or updated. Clicking on the "Default" field next to the "All" category will provide you with the opportunity to install every Cygwin package. Be advised that this will download and install hundreds of megabytes to your computer. The best plan is probably to click on individual categories and install either entire categories or packages from the categories themselves.
The latest net releases of the Cygwin DLL are numbered 1.n.x, where "n" is currently "7" (e.g., 1.7.5). The 1.n.x version numbering refers only to the Cygwin DLL. Individual packages like bash, gcc, less, etc. are released independently of the DLL. The setup.exe utility tracks the versions of all installed components and provides the mechanism for installing or updating everything available from this site for Cygwin.
Once you've installed your desired subset of the Cygwin distribution, setup.exe will remember what you selected so rerunning the program will update your system with any new package releases.
Q: Is there a command-line installer?
http://cygwin.com/setup.exe

Figure 8.1. Cygwin website with install link

2. Go through the installation and, if possible, use default installation settings. There is one mandatory development package that must be installed in addition to the default installation. The package is called `make`. Select the package at the end of the installation. Select `devel`, then package `make` and select it in order to install it (see screenshots below).

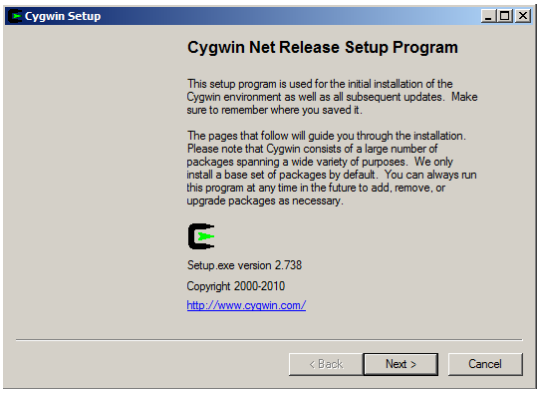

Cygwin Setup
Cygwin Net Release Setup Program
This setup program is used for the initial installation of the Cygwin environment as well as all subsequent updates. Make sure to remember where you saved it.
The pages that follow will guide you through the installation. Please note that Cygwin consists of a large number of packages spanning a wide variety of purposes. We only install a base set of packages by default. You can always run this program at any time in the future to add, remove, or upgrade packages as necessary.
Setup.exe version 2.738
Copyright 2000-2010
http://www.cygwin.com/
< Back
Next >
Cancel

Figure 8.2. Cygwin installation

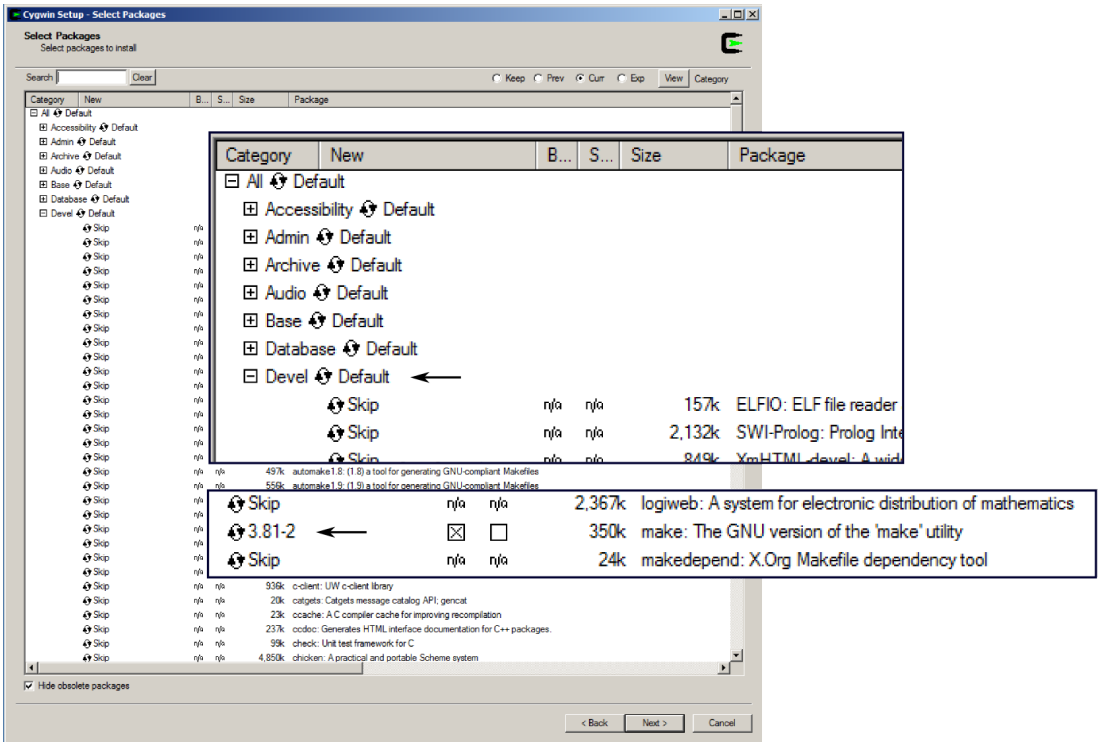


Figure 8.3. Cygwin package selection: Choose package `make`

3. When the Cygwin installation has been finished, download the ziRTK software bundle and unzip it in your Cygwin home folder.

   The home folder is located at: `[Cygwin installation path]\home`

   The default path is: `C:\cygwin\home`

4. Now open the Cygwin shell by clicking either by navigating to the Cygwin start menu entry or by clicking on the Desktop icon.

5. Navigate into the extracted directory. The install script is called "install.sh".

   ```
   cd ziRTK-Win-[VERSION]
   ```

6. Run the install script and go through the guided installation.

   ```
   bash install.sh
   ```

   Alternatively, you can also give executable rights to the install script and run it directly.

   If possible, use default installation paths.

Note

You need not edit your code inside Cygwin, it's just used for compilation and loading your real-time programs onto the HF2 Instrument.

8.1.2. Installation on Linux

Software Requirements

Officially, only Ubunutu 10.04 LTS is supported, but it should be possible to run the tools on any recent Linux distribution. The program GNU make is required to compile the examples. Since the RT tools were compiled on a 32-bit architecture, you need the package `ia32_libs` installed on a 64-bit architecture in order to execute 32-bit programs on a 64-bit architecture. On a debian-based system, both packages can be installed with:

```
sudo apt-get install ia32_libs make
```

Installation Steps

1. Before you begin with the ziRTK installation make sure that the development package "make" is installed.

   ```
   sudo apt-get install make
   ```

2. Extract the ziRTK bundle in a temporary directory.

   ```
   tar xzf ziRTK-[build number]-linux.tar.gz
   ```

3. Navigate into the extracted directory. The install script is called "install.sh".

   ```
   cd ziRTK-Linux-[VERSION]
   ```

4. Run the install script with root rights and go through the guided installation.

   ```
   sudo bash install.sh
   ```

   Alternatively, you can also give executable rights to the install script and run it directly.

   If possible, use default installation paths.

8.1.3. Accessing the Documentation

The developers of Zurich Instruments now happily recommend you to browse remaining examples and reference documentation in HTML format, which duplicates the following section of the user manual. You can find the HTML version of the documentation in

`[INSTALLPATH]/ziRTK/doc/html/index.html`,

which is typically found at

`C:\cygwin\usr\share\zi\ziRTK-XX.XX\doc\html\index.html`

on Windows or

`/opt/zi/ziRTK/doc/html/index.html`

on Linux.

8.2. Real-Time Option Reference Manual

This documentation is for the Zurich Instruments' Real-time Option which enables users to write custom real-time programs that run on an HF2 Instrument.

The recommended way to browse this documentation is in HTML format, see the previous section for help to find it.

The documentation consists of:

- getting started which describes the necessary steps to compile and load a real-time program to an HF2 Instrument,
- a simple example that explains the structure of an RTK program,
- other example programs that demonstrate the most important concepts for controlling and configuring a real-time program,
- a tips and tricks section to get the best performance from your programs,
- a brief description of the most important development tools,
- an API Reference, that can either be browsed conceptually by module or as a complete list of available functions.

8.2.1. Getting Started

This quick-start section demonstrates how to:

- configure the real-time development environment,
- compile C source code,
- run tools on the binary to view memory usage,
- load the binary to the HF2 Instrument.

Please note that the exact shell output and the paths may differ from your installation.

Configuring the RT Development Environment

In order to use the development tools available in your RTK installation you have to configure your shell's `PATH` to include them. This can be done by sourcing the bash file included in the RTK installation directory, in your terminal type:

```
source [INSTALLPATH]/ziRTK-[VERSION]/settings.sh
```

Typically, in Windows (using cygwin) `INSTALLPATH` is `/usr/share/` and in Linux `INSTALLPATH` is `/opt/zi/`.

In order to assert that the `PATH` has been set correctly, try locating the compiler `mb-gcc` with the `which` program and displaying its help message:

```
$ which mb-gcc
/usr/share/zi/ziRTK/tools/bin/mb-gcc.exe
$ mb-gcc --help
...
```

To automatically have the development tools available when you start a new bash shell you can add the statements in `settings.sh` to your own bash configuration file `.bashrc`, located in your home directory.

Note:

If you're using Windows you can find your `.bashrc` file (in a typical cygwin installation) at

```
C:\cygwin\home\[USERNAME]\.bashrc
```

and the `settings.sh` file (typically) at

```
C:\cygwin\usr\share\zi\ziRTK-[VERSION]\settings.sh.
```

Compiling and Running the Examples

The Real-time option comes with a set of examples to demonstrate the main concepts of programming within the real-time environment. They are installed in your RTK example directory (by default `~/ziRTKExamples/`) during installation but can also be found in the examples directory of the ziRTK bundle downloaded from Zurich Instruments' download page, .

There are two subdirectories in the RTK example directory (by default `ziRTKExamples`):

- `examples`, which contains some RT examples that are ready to compile and run,
- `skeleton`, a minimalistic example that can be used as a template for your own programs.

To compile and load an example, perform the following steps:

- Configure your path, as described above.
- In a terminal (under Windows, start Cygwin) navigate to a directory of one of the RTK examples, for example, `AuxInToAuxOut` .
- Run `make` in the example directory by typing

  ```
  make
  ```

  This runs the make program using the `Makefile` found in the example's directory. The `Makefile` describes rules how the binary should be compiled from the source code. You should see output similar to:

  ```
  $ make
  mb-gcc -specs=/opt/zi/ziRTK/lib/microblaze/specs
  -B/opt/zi/ziRTK/tools/bin/ -mno-xl-soft-mul -mxl-barrel-shift
  -mxl-pattern-compare -mhard-float -mcpu=v7.10.b -O3
  -fno-strict-aliasing -mxl-float-convert -mxl-float-sqrt
  -fsingle-precision-constant -Winline -Wall -Wextra -std=gnu99
  -I./. -I/opt/zi/ziRTK/include -c -o obj/AuxInToAuxOut.o
  AuxInToAuxOut.c
  mb-gcc -specs=/opt/zi/ziRTK/lib/microblaze/specs
  -B/opt/zi/ziRTK/lib/microblaze/ -L/opt/zi/ziRTK/lib/microblaze
  -T/opt/zi/ziRTK/lib/microblaze/linker_script_mb_standard.ld
  -Wl,--as-needed -Wl,--start-group
  /opt/zi/ziRTK/lib/microblaze/libziRTKmb.a obj/AuxInToAuxOut.o
   -Wl,--end-group -Wl,--no-as-needed -lm -o AuxInToAuxOut.mem.elf
   Determining Size of ELF File **************
  mb-size AuxInToAuxOut.mem.elf
     text    data     bss     dec     hex filename
    41650     372    3096   45118    b03e AuxInToAuxOut.mem.elf

   Generating mem file **************
  data2mem -bd AuxInToAuxOut.mem.elf -d -o m AuxInToAuxOut.mem
  ```

  Invoking the `make` command compiles, links, generates the binary `.mem`-file and runs some statistics on the binary. The next section tells you how to interpret the memory statistics.

To remove the generated files execute the command `make clean`:

```
$ make clean
rm -f obj/AuxInToAuxOut.o
rm -f AuxInToAuxOut.mem
rm -f AuxInToAuxOut.mem.elf
rm -f deps/AuxInToAuxOut.d
rm -rf obj
rm -rf deps
```

Analyze Memory Usage

The `ls -l` command lists files and reports information about them (like the size). Don't be shocked by the size of your elf file, the actual memory consumption is much smaller. Running `mb-size` on the generated ELF (Executable and Linkable Format) file informs you on the actual memory consumption.

```
$ ls -l AuxInToAuxOut.mem.elf
-rwxr-xr-x 1 user users 112957 2012-11-23 13:02 AuxInToAuxOut.mem.elf
$ mb-size AuxInToAuxOut.mem.elf
   text    data     bss     dec     hex filename
  41650     372    3096   45118    b03e AuxInToAuxOut.mem.elf
$ mb-strip AuxInToAuxOut.mem.elf
$ ls -l AuxInToAuxOut.mem.elf
-rwxr-xr-x 1 user users 43312 2012-11-23 13:03 AuxInToAuxOut.mem.elf
```

The output figures of `mb-size` are:

- `text` size of the program code. These are the actual commands. Note that single-value constants are often integrated in the text.
- `data` amount of data. These are constants like arrays of numbers and strings.
- `bss` the amount of data RAM that is going to be used uninitialized. This is needed for variables to compute with.
- `dec` the total memory usage in a decimal number
- `hex` the total memory usage in a hex number

To be sure that you do not include unneeded information, i.e. debugging information, in the binary you can use `mb-strip`. `mb-strip` strips all debugging information and symbol tables from an elf or object file. In this case you see that the file size is unaffected by running `mb-strip` and everything is OK.

You can find a short description of the other most important tools here .

Loading the Binary to the HF2 Instrument

To load the compiled example to your HF2 Instrument on the command line run `make` specifying the target `prog`

```
make prog
```

This invokes the command line program `zirtkprog` to load the program.

To load the compiled example from ziControl select to 'Realtime' tab and click the 'Program' button. Note, however, the program has to be compiled from the command line.

8.2.2. Tips and Tricks

- Unless absolutely necessary, it is highly discouraged to use the `double` data type. The hardware floating-point unit only supports single-precision (`float`). Therefore using double will make your code ~10-times bigger and 10-times slower.
- Most maths functions exist in a double and a float variant. The float variants have an "f" appended. E.g. `cos(double x)` and `cosf(float x)`. For optimal performance, make sure that you always use the float variants. In order to compare the timings of calling `sin()` versus `sinf()` for example, see the SpeedTest example .
- Configure parameters of your real-time program at run-time to avoid hard-coding and recompiling your program, see the UserRegs and MultipleParameters examples.
- View log messages sent to the PC with the ziRTKPrintf() function in the Realtime tab of ziControl by enabling the Log messages button. Log messages can be viewed on the command line with the program `zirtkcat`.
- Use multiple triggers and test which trigger has fired for advanced control of your program, see the MultipleTriggers example .
- Check that you're not dropping samples when using a sample-based trigger such as ziRTKAddDemodSampleTrigger() by ensuring that your program is occupying (slightly) less than 100% CPU. View CPU usage in the realtime tab of ziControl or via the node `/devX/cpus/0/workload`.
- Test how fast your ziRTKLoop() can run by calling it via a clock trigger (ziRTKAddClockTrigger()) with delay 0 and measuring its speed with code similar to that in the SpeedTest example .
- Be aware that calling functions that change instrument settings, e.g., ziRTKDIOSetOutput(), or send data to the PC, e.g., ziRTKPrintf(), can cause a large amount of data to be sent between the real-time program and the PC. In extreme cases this can overload ziServer and cause the real-time program to stop running properly. In order to minimise this:
  - When possible use a 'NoUpdate' version of the function, i.e., use ziRTKDIOSetOutputNoUpdate() instead of ziRTKDIOSetOutput().
  - Avoid a large number of calls (at full loop speed) to ziRTKPrintf(), ziRTKUserRegGet() and ziRTKUserRegSet().

8.2.3. Example Real-time Programs

The source code and Makefiles for the examples can be found either in the real-time examples directory that is created during the installation of RTK or in the 'examples' folder of the RTK download bundle avaible from .

- Program structure: A simple program demonstrating ziRTKInit() and ziRTKLoop()
- User registers: Data transfer between the PC and the RT environment
- Triggers: Write values to an auxilliary output via a clock trigger
- Triggers: Control loop behaviour with multiple triggers
- User registers: Control loop behaviour and multiple program parameters with user registers
- Loop Structure: Record demodulator data upon a DIO trigger using a finite state machine
- Performance: Calculate the update rate of a real-time program

8.2.4. Program structure: A simple program demonstrating ziRTKInit() and ziRTKLoop()

In the real-time programming environment there are two functions available to the user which define how a real-time program runs: ziRTKInit() and ziRTKLoop(). The main() function is not directly available to the user, it is implemented elsewhere and manages low-level functionality (such as the triggers to ziRTKLoop()). The user must define both functions, although either function can be empty.

ziRTKInit() is called once upon loading the real-time program onto the HF2, it is used to initialise the program and HF2. It also defines which triggers should repeatedly cause ziRTKLoop() to be called. ziRTKInit() can configure ziRTKLoop() to run, for example, every time a new demodulator sample output is available (ziRTKAddDemodSampleTrigger()) or every time a trigger signal is sent on the HF2's DIO (ziRTKAddDIOSampleTrigger()). ziRTKInit() can also specify that ziRTKLoop() be called with a fixed time delay (ziRTKAddClockTrigger()), or just as fast as possible (ziRTKAddClockTrigger() with delay 0), so that ziRTKLoop() just gets called back-to-back with minimal delay in between.

For help compiling the example see the getting started section .

```
/* $Rev: 14829 $ $Date: 2012-11-23 15:23:59 +0100 (Fri, 23 Nov 2012) $
 *
 * Description:
 *
 *  This example copies the values from auxilliary inputs 0 and 1
 *  directly to the auxilliary outputs 0 and 1. ziRTKLoop() is defined
 *  to be called every time a new auxilliary input sample arrives via
 *  calling the function ziRTKAddAuxInSampleTrigger() to ziRTKInit().
 *
 *  The averaging performed at the auxilliary input can be changed by
 *  modifying the value of the 0th user register. This can be done in
 *  the Realtime tab of ziControl.
 *
 */

#include <math.h>
#include <ziRTK.h>

AuxInSample Sample;
float Ch0, Ch1;

unsigned int Reg, TS=0, PrevTS=0;
unsigned int Averaging = 10000;

void ziRTKInit() {

  ziRTKAddAuxInSampleTrigger( 0, NULL );
  ziRTKAuxInSetAveraging( 0, Averaging );

  // Set auxout 0 and 1 to manual
  ziRTKAuxOutSetOutputSelect( 0, -1 );
  ziRTKAuxOutSetOutputSelect( 1, -1 );

}

void ziRTKLoop() {

  // Get sample from auxin
  ziRTKAuxInGetSample( 0, &Sample );

  ziRTKAuxInSampleGetValue( &Sample, 0, &Ch0 );
  ziRTKAuxInSampleGetValue( &Sample, 1, &Ch1 );
```

```
  ziRTKAuxOutSetOffset( 0, Ch0 );
  ziRTKAuxOutSetOffset( 1, Ch1 );

  ziRTKUserRegGet( 0, &Reg );

  TS = ziRTKGetTimeStamp32();

  if( Reg != Averaging ){
    Averaging = Reg;
    ziRTKAuxInSetAveraging( 0, Averaging );
  }

  //ziRTKPrintf("dT = %08u, UserReg0 = %u\n", ( TS - PrevTS ), Reg);

  PrevTS = TS;
}
```

8.2.5. User registers: Data transfer between the PC and the RT environment

```
/* $Id: UserRegs.c 14843 2012-11-26 14:53:17Z danielw $
 *
 * Description:
 *
 * This example demonstrates how to modify a parameter of a running
 * real-time program. This is achieved via so-called user registers
 * (0...63) which are dedicated 32-bit memory slots for a user to
 * transfer data between the PC and a real-time program.
 *
 * Here we get the R value from a demodulator and output it on
 * auxilliary output 0, scaled by a user-specified floating-point
 * factor obtained from user register 0.
 *
 * Test this program:
 *
 * Connect a feedback cable from signal input 0 to signal output 0 and
 * load the configuration file in this folder in the Save tab of
 * ziControl. In the Realtime tab in ziControl load the binary
 * UserRegs.mem, you should see the counter increment in the UserReg 1
 * in the 'Decimal/Hex' field. Enable the log and modify the
 * scaleFactor parameter in the 'Float Value' field of UserReg 0. You
 * can observe that the scaleFactor printed in the log is modified,
 * and, by going to the Auxilliary I/0 Tab, the value of R written to
 * auxilliary output 0 is also modified accordingly.
 *
 * Note:
 *
 * Calling ziRTKLoop() at a very high frequency (e.g., by increasing
 * the demodulator's output rate) would cause a large amount of data
 * to be sent between the real-time program and the PC (due to
 * ziRTKUserRegGet(), ziRTKUserRegSet() and ziRTKPrintf(). In extreme
 * cases this can overload ziServer and cause the real-time program to
 * stop running properly.
 *
 */

#include <math.h>
#include <ziRTK.h>

// helper to obtain floating point values from user registers
typedef union {
    unsigned int as_uint32;
```

```
    float as_float;
} union_uint32_float_t;

unsigned int demodIndex = 0;
unsigned int auxOutIndex = 0;

void ziRTKInit() {

    // Define a demodulator sample trigger
    ziRTKAddDemodSampleTrigger( demodIndex, NULL );
    // Use a slow demodulator output rate
    ziRTKDemodSetRate( demodIndex, 10.0f );

    // Set auxilliary output 0 to manual
    ziRTKAuxOutSetOutputSelect( auxOutIndex, -1 );

}

DemodSample SampleD;
float R = 0.0f;
float scaleFactor = 1.0f;
unsigned int counter = 0;

void ziRTKLoop() {

    // Update configuration. Note we have to do a reinterpret typecast
    // in order to retrieve a floating point value from a user
    // register. Necessary since ziRTKUserRegGet() retrieves an
    // unsigned integer but we want the floating-point number with the
    // same bit representation.
    union_uint32_float_t userreg;
    ziRTKUserRegGet( 0, &userreg.as_uint32 );
    scaleFactor = userreg.as_float;

    // Note if scaleFactor was an integer and not a floating point
    // value, it would be possible to just use:
    // unsigned int scaleFactor;
    // ziRTKUserRegGet( 0, &scaleFactor );

    // Get the demodulator R value
    ziRTKDemodGetSample( demodIndex, &SampleD );
    ziRTKDemodSampleGetCompR( &SampleD, &R );

    // Write the scaled value to the auxilliary output
    ziRTKAuxOutSetOffset( auxOutIndex, scaleFactor*R );

    // Also print the scaled value to the log
    ziRTKPrintf("\n R: %e, scaleFactor: %e\n",R,scaleFactor);

    // Write a counter to user register 1
    ziRTKUserRegSet( 1, counter);
    counter++;

}
```

8.2.6. Triggers: Write values to an auxilliary output via a clock trigger

```
/* $Rev: 937 $ $Date: 2009-01-09 20:19:12 +0100 (Fri, 09 Jan 2009) $
 *
 * Description:
 *
```

```
 * This example generates a sine wave (sinRet) on the auxilliary
 * output channel 0 and sets the frequency of oscillator 0 according
 * to fabs(sinRet)*1000000 with a fixed delay interval. The delay is
 * defined using a clock trigger in ziRTKInit().
 *
 * Note:
 *
 * Calling ziRTKLoop() at a very high frequency (e.g., by increasing
 * the demodulator's output rate) would cause a large amount of data
 * to be sent between the real-time program and the PC (due to
 * ziRTKAuxOutSetOffset()). In extreme cases this can overload
 * ziServer and cause the real-time program to stop running
 * properly. Please see the note below about the 'NoUpdate' version of
 * ziRTKAuxOutSetOffset().
 *
 */

#include <math.h>
#include <ziRTK.h>


void ziRTKInit() {

  // Define the delay for the clock trigger (in milliseconds)
  unsigned int clockTriggerDelay = 200;
  ziRTKAddClockTrigger( clockTriggerDelay, NULL );

  // Set auxilliary output channel 0 to manual
  ziRTKAuxOutSetOutputSelect( 1, -1 );

}

unsigned int counter = 0;
float phaseAcc = 0.;

void ziRTKLoop() {

  const float sinRet = sinf( phaseAcc );

  // Output on auxilliary outputs using with a 10 V amplitude
  // Auxilliary output range is -10 to 10 volts, scale accordingly
  ziRTKAuxOutSetOffset( 0, sinRet * 10. );
  // Note: if ziServer is getting overloaded by a lot of data
  // transfer, try using the no update version:
  // ziRTKAuxOutSetOffsetNoUpdate(), which sets the
  // values in hardware but doesn't communicate the value via USB to
  // the ziServer

  ziRTKOscSetFreq( 0, fabs( sinRet ) * 1000000 );

  // Accumulate phase
  phaseAcc += 0.001;
  if( phaseAcc > (float) 2. * 3.14159265 )
    phaseAcc-= (float) 2. * 3.14159265;

  // Say something every 1000 times
  if( counter % 1000 == 0 ) {
      ziRTKPrintf( "phaseAcc: %f, counter: %d \n",
                   phaseAcc, counter );
    }

  counter++;

}
```

8.2.7. Triggers: Control loop behaviour with multiple triggers

```
/* $Rev: 14843 $ $Date: 2012-11-26 15:53:17 +0100 (Mon, 26 Nov 2012) $
 *
 * Description:
 *
 * This example demonstrates how to use multiple triggers in order to
 * perform two actions. The two actions are:
 *
 * 1. When a new sample is available from the demodulators ziRTKLoop()
 * obtains the demodulator's R and Theta values and writes them to the
 * first two auxilliary output channels.
 *
 * 2. When a new sample is available from the first auxilliary input
 * channel it writes the value to the digital output.
 *
 * Additionally, either the value of R and theta or the auxilliary
 * input sample is printed to the log. This be viewed either in the
 * log messages of ziControl's Realtime tab or in a terminal using the
 * program zirtkcat.
 *
 * This is achieved by adding demodulator and auxilliary input
 * triggers in ziRTKInit(). ziRTKLoop() checks which of the triggers
 * is active and performs the relevant action.
 *
 * Note: Calling ziRTKLoop() at a very high frequency (e.g., by using
 * a very high demodulator output rate) would cause a large amount of
 * data to be sent between the real-time program and the PC (due to
 * ziRTKPrintf()). In extreme cases this can overload ziServer and
 * cause the real-time program to stop running properly.
 *
 */

#include <math.h>
#include <ziRTK.h>

ziTriggerId DemodTrigger;
ziTriggerId AuxInTrigger;

void ziRTKInit() {

  // Define when ziRTKLoop() should be called
  ziRTKAddDemodSampleTrigger( 0, &DemodTrigger );
  ziRTKAddAuxInSampleTrigger( 0, &AuxInTrigger );

  // Set a slow demodulator output rate
  ziRTKDemodSetRate( 0, 1 );

  // Set auxilliary output channels 0 and 1 to manual
  ziRTKAuxOutSetOutputSelect( 0, -1 );
  ziRTKAuxOutSetOutputSelect( 1, -1 );

  // Set the DIO to drive outputs
  ziRTKDIOSetDrive( 0, 3 );

}

DemodSample SampleD;

float R;
float Theta;

AuxInSample SampleAI;
float AuxInValue;

unsigned int IsTriggered;
```

```
void ziRTKLoop() {

  ziRTKWriteString( "Start of ziRTKLoop\n" );

  ziRTKTestTrigger( DemodTrigger, &IsTriggered );

  if( IsTriggered ) {
    // Retrieve the sample of demodulator 0
    ziRTKDemodGetSample( 0, &SampleD );

    // Retrieve the magnitude of the demodulator 0 sample.
    // All voltages in V
    ziRTKDemodSampleGetCompR( &SampleD, &R );

    // Retrieve the phase
    ziRTKDemodSampleGetTheta( &SampleD, &Theta );

    // Output the magnitude on stdout
    ziRTKPrintf( "Demod. R:   %f\n", R );

    // Write R to on auxilliary output channel 0
    ziRTKAuxOutSetOffset( 0, R );
    // Write Theta on auxilliary output channel 1
    // Auxilliary output range is -10 to 10; scale Theta accordingly
    ziRTKAuxOutSetOffset( 1, Theta / 18 );
  }

  ziRTKTestTrigger( AuxInTrigger, &IsTriggered );

  if( IsTriggered ) {

    // Retrieve the sample from auxilliary input 0
    ziRTKAuxInGetSample( 0, &SampleAI );

    // Retrieve the value of the sample from channel 0
    ziRTKAuxInSampleGetValue( &SampleAI, 0, &AuxInValue );

    ziRTKPrintf( "AuxInValue: %f\n", AuxInValue );

    // Write the auxilliary output value to the digital output.
    // Auxilliary input range is -10 to 10; add 10 to obtain a
    // positive value.
    // Digital output range is 0 to 2^16; multiply by 2^16 to attain
    // an integer value in that range.
    ziRTKDIOSetOutput( 0, ( AuxInValue + 10 ) * 0x7fff );
  }

}
```

8.2.8. User registers: Control loop behaviour and multiple program parameters with user registers

```
/* $Id: MultipleParameters.c 14878 2012-11-27 16:40:41Z danielw $
 *
 * Description:
 *
 * This program obtains demodulator 0's R value, scales and offsets it
 * and writes it to an auxilliary output. ziRTKLoop() is called on a
 * demod trigger. The scale, offset and auxilliary channel are all
 * configurable via user registers. The mode (idle,compute,debug) of
 * the program can also be controlled by a user register.
```

```
 *
 * Test this program:
 *
 * Connect a feedback cable from signal output 1 to signal input 1 and
 * load the configuration file MultipleParameters.zicfg in this
 * folder. You should obtain a near constant value on the R value of
 * demodulator 0, it is this value we will write to the auxilliary
 * outputs.
 *
 * Now go to the Realtime tab in ziControl and first load the xml file
 * containing the user register names, RegisterNames.xml, by clicking
 * on the 'Reg Name' pull-down menu and selecting 'Open...'. You
 * should see the first 5 user register names get populated.
 *
 * Now load the real-time program's binary file
 * MultipleParameters.mem. The CONTROL register is used to define the
 * program's behaviour, if no bits of the CONTROL register are set,
 * then the program is idle. The bits of CONTROL have the following
 * meaning:
 *
 * - 1: perform a configuration update
 * - 2: call the Compute() function (compute the average)
 * - 3: show some debugging output
 *
 * Accordingly, setting the CONTROL register to the following values
 * has the described affect:
 * - control=0, idle
 * - control=1, perform config update and then go idle
 * - control=2, compute (average R and write to auxout)
 * - control=3, perform config update once and then compute
 * - control=6, compute and debug (write average R to R_AVERAGE
 *              register)
 * - control=7, perform config update once and then compute and debug
 *              (write average R to R_AVERAGE register) See
 *              ziRTKLoop() for more details.
 *
 *
 * The program's CONTROL is initially set to 2 (compute bit is set),
 * and we can see that the demodulator's R value is being written to
 * auxilliary output channel 0 in the Auxilliary I/0 Tab in ziControl.
 *
 * Now set the CONTROL register to 6, which runs the program in
 * 'compute and debug' mode, now you should see the average value of R
 * update in the 'DEBUG' register's 'Float Value' field.
 *
 * The scale and offset used in the program can be modified in the
 * corresponding register's 'Float Value' field and the auxilliary
 * output channel to be used can be specified in the AUXOUT_CHANNEL
 * registers 'Decimal' field. Send the configuration update to the
 * program by setting the 1st bit of the CONTROL reg as described
 * above.
 *
 */

#include <math.h>
#include <ziRTK.h>

// Define which user registers hold which configuration
#define CONTROL_REG        0
#define AUXOUT_CHANNEL_REG 1
#define SCALE_REG          2
#define OFFSET_REG         3
#define DEBUG_REG          4

// helper to obtain floating point values from user registers
typedef union {
    unsigned int as_uint32;
```

```
    float as_float;
} union_uint32_float_t;

// Globals ////////////////////
// Program parameters
float Scale = 1.0f;  // Scaling factor for the output
float Offset = 0.0f; // Offset for the output
unsigned int AuxOutChannel = 0;

// Helpers for debugging
// Write debugging output DebugPeriodSeconds times per second
float DebugPeriodSeconds = 0.5f;
unsigned int DebugPeriod = 0;

void ziRTKInit() {

  // Add a demod trigger for ziRTKLoop()
  ziRTKAddDemodSampleTrigger( 0, NULL );

  // Set auxout 0 to manual
  ziRTKAuxOutSetOutputSelect( AuxOutChannel, -1 );

  // Zero all user registers for tidiness
  for(int i=0;i<64;i++) {
    ziRTKUserRegSet( i, 0 );
  }

  // Initialise user registers that are used for program parameters
  // with the initial values set in this program.
  // Note that we have to use our union helper union_uint32_float_t to
  // set floating-point values (see also UserRegs example).
  unsigned int initialControl = 2;
  ziRTKUserRegSet( CONTROL_REG, initialControl );
  ziRTKUserRegSet( AUXOUT_CHANNEL_REG, AuxOutChannel );

  union_uint32_float_t userreg;
  userreg.as_float = Scale;
  ziRTKUserRegSet( SCALE_REG, userreg.as_uint32 );
  userreg.as_float = Offset;
  ziRTKUserRegSet( OFFSET_REG, userreg.as_uint32 );

  float demodRate = 0.0f;
  ziRTKDemodGetRate( 0, &demodRate );
  DebugPeriod = (unsigned int)(ceil(DebugPeriodSeconds*demodRate));

}

// Retrieve a configuration update from the user registers
inline void UpdateConfig(void) {

  // Note that we have to use our union helper union_uint32_float_t to
  // set floating-point values (see also UserRegs example).
  union_uint32_float_t userreg;

  ziRTKUserRegGet( SCALE_REG, &userreg.as_uint32 );
  Scale  = userreg.as_float;

  ziRTKUserRegGet( OFFSET_REG, &userreg.as_uint32 );
  Offset = userreg.as_float;

  ziRTKUserRegGet( AUXOUT_CHANNEL_REG, &AuxOutChannel );
  // Set the auxilliary output channel to manual
  ziRTKAuxOutSetOutputSelect( AuxOutChannel, -1 );

  // Tell the user the config update via the log
  ziRTKPrintf(
    "\nUpdated config: AuxOutChannel: %d, Scale: %e, Offset: %e \n",
```

```
    AuxOutChannel, Scale, Offset );

  float demodRate = 0.0f;
  ziRTKDemodGetRate( 0, &demodRate );
  DebugPeriod = (unsigned int)(ceil(DebugPeriodSeconds*demodRate));

}

float result = 0.0f;

// Perform our computation on a new demod sample
inline void Compute(void) {

  // Get the R value of the first demodulator
  DemodSample SampleD;
  ziRTKDemodGetSample( 0, &SampleD );
  float R = 0.0f;
  ziRTKDemodSampleGetCompR( &SampleD, &R );

  // Perform a computation on R here..
  // R = ...

  // Scale and offset
  result = Scale*R + Offset;

  // Write the result to the auxilliary output
  ziRTKAuxOutSetOffsetNoUpdate( AuxOutChannel, result );

}

inline void Debug(void) {

  static unsigned int debugCount = 0;
  if( debugCount > DebugPeriod ) {
    // Write the current output to a user register
    union_uint32_float_t userreg;
    userreg.as_float = result;
    ziRTKUserRegSet( DEBUG_REG, userreg.as_uint32 );
    debugCount = 0;
  }
  debugCount++;

}

// The main loop
void ziRTKLoop() {

  // Get control register
  unsigned int control;
  ziRTKUserRegGet( CONTROL_REG, &control );

  // Update configuration?
  if (control & 0x0001) {
    UpdateConfig();
    ziRTKUserRegSet( CONTROL_REG, control & 0xfffffffe );
  }

  // Compute?
  if (control & 0x0002) {
    Compute();
  }

  // Debug?
  if (control & 0x0004) {
    Debug();
  }
```

```
}
```

8.2.9. Loop Structure: Record demodulator data upon a DIO trigger using a finite state machine

```
/* $Rev: 14843 $ $Date: 2012-11-26 15:53:17 +0100 (Mon, 26 Nov 2012) $
 *
 * Description:
 *
 * Write a burst of demodulator data to an auxilliary output.
 *
 * Upon being triggered externally by a value written on the HF2's
 * digital input, this program writes RecordingTime milliseconds of
 * data from the demodulators to an auxilliary output.
 *
 * The recording behaviour is implemented in ziRTKLoop() as a finite
 * state machine which acts on fresh demodulator samples as defined by
 * the demodulator trigger in ziRTKLoop().
 *
 * Test this program:
 *
 * After loading DemodRecorder.mem, go to the Auxilliary Outputs tab
 * in ziControl, enable the drive button of the lower Digital I/O bits
 * and change the lowest bit (DIO_PIN) of the input. In Auxilliary
 * Output 1's Value field you can see that the demodulator's magnitude
 * is written for a duration of RecordingTime milliseconds.
 *
 */

#include <math.h>
#include <ziRTK.h>

void ziRTKInit() {

  ziRTKAddDemodSampleTrigger( 0, NULL );

  // Set auxilliary output channel 0 to manual
  ziRTKAuxOutSetOutputSelect( 0, -1 );

  // Set DIO not to drive outputs
  ziRTKDIOSetDrive( 0, 0 );

}

// DIO pin which will be used as trigger source
// (may be set by the user)
#define DIO_PIN           0

// duration of the recording time in Milliseconds
// (may be set by the user)
const unsigned int RecordingTime = 2000;

// States of the finite state machine
#define STATE_IDLE        0
#define STATE_RECORDING   1
#define STATE_WAIT        2

// Initialize state of the finite state machine
unsigned int State = STATE_IDLE;

// Will hold the timestamp of the first sample in recording window
unsigned int StartTS = 0;
```

```
// Will be '1' if the trigger dio pin is in the active state
unsigned int TriggerPin = 0;

DemodSample SampleD;

DIOSample SampleDIO;
unsigned int DIOBits;

// Counter used for counting the samples being recorded
// just for testing!
unsigned int Count = 0;

void ziRTKLoop() {

  // Read the DIO sample
  ziRTKDIOGetSample( 0, &SampleDIO );
  ziRTKDIOSampleGetBits( &SampleDIO, &DIOBits );

  // Check if the pin is low (active state)
  TriggerPin = ( ( DIOBits & ( 1 << DIO_PIN ) ) == 0 ) ? 1 : 0;

  if( State == STATE_IDLE ) {

    ziRTKAuxOutSetOffset( 0, 0 );

    // Wait for the trigger pin going to active state
    if( TriggerPin == 1 )
      State = STATE_RECORDING;

    StartTS = 0;
    Count = 0;

  }

  if( State == STATE_RECORDING ) {
    // Record as long as the timestamp of the demodsample lies within
    // the recording window

    // Retrieve the sample from demodulator 0
    ziRTKDemodGetSample( 0, &SampleD );

    unsigned int CurrentTS;
    ziRTKDemodSampleGetTimeStamp32( &SampleD, &CurrentTS );

    if( StartTS == 0 ) {
      // Set the start timestamp if the current sample is the first
      // recorded sample ( StartTS == 0 )
      StartTS = CurrentTS;
    }
    else {
      // Check if the time recorded is bigger then the required
      // recording time
      // If so advance state to STATE_WAIT

      if( ( CurrentTS - StartTS )
          / ( ZIRTK_HF2_SAMPLERATE / 1000 ) > RecordingTime )
        State = STATE_WAIT;

    }

    if( State == STATE_RECORDING ) {
      // Current timestamp lies within time window
      // (State has not been advanced in check)

      // Do something with the samples here!
```

```
      // For testing purposes printf the counter and increment it
      ziRTKPrintf("Count: %d\n", Count );
      Count++;

      float Mag;
      ziRTKDemodSampleGetCompR( &SampleD, &Mag );
      Mag = Mag * 10;             // Scale the Mag to 0V..10V
      if( Mag > 10 ) Mag = 10;    // Clip to 10V

      // Output on auxilliary output
      ziRTKAuxOutSetOffset( 0, Mag );
    }

  }

  if( State == STATE_WAIT ) {

    ziRTKAuxOutSetOffset( 0, 0 );

    // Wait until the trigger pin returns to inactive state to return
    // to beginning
    if( TriggerPin == 0 )
      State = STATE_IDLE;

  }

}
```

8.2.10. Performance: Calculate the update rate of a real-time program

```
/* $Id: SpeedTest.c 14878 2012-11-27 16:40:41Z danielw $
 *
 * Description:
 *
 * This example shows how to check the update rate of ziRTKLoop().
 *
 * Test this program:
 *
 * Load the program SpeedTest.mem and the RegisterNames.xml file in
 * the Realtime tab of ziControl. The approximate time taken and
 * update rate of ziRTKLoop() is shown in the user registers 1 and 2.
 * Change the decimal value of the Control register to modify which
 * code is executed. We see that the floating point version of sin()
 * runs considerably faster.
 *
 */

#include <math.h>
#include "ziRTK.h"

// Define user registers
#define CONTROL_REG               0
#define PERFORMANCE_DT_REG        1
#define PERFORMANCE_RATE_REG      2

// Helper for writing floats to user registers
typedef union {
  unsigned int as_uint;
  float as_float;
} union_uint_float;
```

```
// Globals ////////////////////////////////////////////////////////

// The delay (milliseconds) for the clock trigger.
// Delay 0 results in continual triggering.
unsigned int ziRTKLoopTimeDelay = 0;

// Variables for performance monitoring:
// Define how often we should calculate and update the performance
unsigned int PerformanceRefreshPeriodSeconds = 1.0f;

// Calculate and update performance after this many counts.
// After the first performance calculation this is adjusted according
// to PerformanceRefreshPeriodSeconds
unsigned int PerformanceRefreshPeriod = 10000;

// Counter to test when to calculate and update performance
unsigned int PerformanceCount = 0UI;

// Store the timestamp to calculate performance
unsigned int PerformanceLastTs = 0UI;


void ziRTKInit() {

  // Define a clock trigger
  const unsigned int timerDelayMicroSeconds = ziRTKLoopTimeDelay;
  ziRTKAddClockTrigger( timerDelayMicroSeconds, NULL );

  // Initialise RTK user registers to zero
  for(unsigned int i=0;i<64;i++) {
    ziRTKUserRegSet( i, 0 );
  }

  PerformanceLastTs = ziRTKGetTimeStamp32();

}

inline void Performance(void) {

  if( PerformanceCount > PerformanceRefreshPeriod ) {

    // Get the current timestamp from the HF2 in ticks (integer)
    const unsigned int ts = ziRTKGetTimeStamp32();

    // Calculate the total time taken and convert it from ticks to
    // seconds by dividing by ZIRTK_HF2_SAMPLERATE
    const float dtPeriod =
      (float)(ts - PerformanceLastTs)/(float)(ZIRTK_HF2_SAMPLERATE);

    // Get the time taken for one loop
    const float dt = dtPeriod/PerformanceRefreshPeriod;
    // and calculate the rate
    const float rate = 1.0F/dt;

    // Adjust timing to run performance calculation approx every
    // PerformanceRefreshPeriodSeconds
    PerformanceRefreshPeriod = PerformanceRefreshPeriodSeconds/dt;

    // See example UserRegs for an explanation
    union_uint_float userreg;

    // Send rate and dt to the PC via user registers
    userreg.as_float = rate;
    ziRTKUserRegSet( PERFORMANCE_RATE_REG, userreg.as_uint );

    userreg.as_float = dt;
    ziRTKUserRegSet( PERFORMANCE_DT_REG, userreg.as_uint );
```

```
    PerformanceCount = 0UI;
    PerformanceLastTs = ziRTKGetTimeStamp32();

  }

  PerformanceCount++;

}

void ziRTKLoop(void) {

  // Get the control register which determines which function we want
  // to calculate
  unsigned int control = 0;
  ziRTKUserRegGet( CONTROL_REG, &control );

  // Declare volatile to prevent the compiler from performing
  // optimisation that would distort our timings
  volatile float f = 0.0f;
  volatile float g = 1.23f;

  switch( control ) {
  case 0:
    // The normal version of sin()
    f = sin(g);
    break;
  default:
    // The floating point version of sin()
    f = sinf(g);
  }

  // Check and calculate(?) the current performance of this program
  Performance();

}
```

8.2.11. Development Tools

This section briefly documents the most important command-line tools delivered with the real-time programming environment. They can be found in

```
[INSTALLPATH]/ziRTK/tools/bin/.
```

For more information please see the Xilinx 'Embedded System Tools Reference Manual'.

- **`zirtkprog`** - loads binary real-time programs (`.mem` files) from the PC to an HF2 Instrument. Note, that it's also possible to load a program via the 'Realtime' tab in ziControl.
- **`zirtkcat`** - displays the output ziRTKPrintf of a real-time program running on an HF2 Instrument. Note, that it's also possible to view the output of a real-time program in the 'Realtime' tab in ziControl.
- **`mb-gcc`** - the cross-compiler for C programs. It generates an `.elf` file from `.c` source files.
- **`data2mem`** - creates a `.mem` file from an `.elf` file.
- **`mb-size`** - displays size information of a `.mem` file.
- **`mb-readelf`** - displays detailed information about the `.elf` file.

Chapter 9. Specifications

Important

Unless otherwise stated, all specifications apply after 30 minutes of instrument warm-up.

Important

Important changes in the specification parameters are explicitly noted in the Revision History.

9.1. General Specifications

Table 9.1. General and storage

Parameter	min	typ	max
storage temperature	-25 °C	-	65 °C
storage relative humidity (non-condensing)	-	-	95%
operating temperature	5 °C	-	40 °C
operating relative humidity (non-condensing)	-	-	90%
specification temperature	18 °C	-	28 °C
power consumption	-	-	60 W
power supply	110-120/220-240 V, 50/60 Hz		
power supply Japan	requires external 100 V to 110 V transformer (50/60 Hz) for operation according to specification		
environmental policy	RoHS compliant		
dimensions with handles and feet	45 x 34 x 10 cm, 17.7 x 13.6 x 4.0 inch, 19 inch rack compatible		
weight	6.2 kg		
recommended calibration interval	2 years		

Table 9.2. Maximum ratings

Parameter	min	typ	max
damage threshold HF inputs (Input 1, Input 2)	-5 V	-	5 V
damage threshold HF outputs (Output 1, Output 2)	-12 V	-	12 V
damage threshold Add inputs (Add 1, Add 2)	-12 V	-	12 V
damage threshold Sync output (Sync 1, Sync 2)	-12 V	-	12 V
damage threshold auxiliary outputs	-12 V	-	12 V
damage threshold auxiliary inputs	-12 V	-	12 V
damage threshold digital I/O (including DIO 0 and DIO 1 BNC connectors)	0 V	-	5 V
damage threshold Clock input	0 V	-	5 V

Table 9.3. Host system requirements

Parameter	Description
supported Windows operating systems	32-bit and 64-bit versions of XP, Vista, Windows 7, Windows 8
supported Linux distribution	32-bit and 64-bit of Linux, Ubuntu 10.04 LTS (i386, AMD64), 64-bit systems require the IA32 extension (install with `sudo apt-get install ia32-libs`)

Parameter	Description
minimum processor requirements	AMD K8 (Athlon 64), Intel Pentium 4
minimum memory requirements	1 GB RAM, 2GB recommended
list of supported processors (requiring SSE2)	AMD K8 (Athlon 64, Sempron 64, Turion 64, etc.), AMD Phenom, Intel Pentium 4, Xeon Celeron, Celeron D, Pentium M, Celeron M, Core, Core 2, Core i5, Core i7, Atom

Table 9.4. Maximum sample readout rate

Active demodulators	Maximum sample readout rate	Comments
1	460 kSamples/s	to achieve highest rates, it is advised to remove all other data transfer that loads the USB - it is recommended to check the sample loss flag (in the status tab) from time to time when using high readout rate settings
2 - 3	230 kSamples/s	
4 - 6	115 kSamples/s	
7 - 8	57 kSamples/s	

9.2. Analog Interface Specifications

Table 9.5. HF signal inputs

Parameter	min	typ	max
connectors	front-panel single-ended/differential BNC		
input impedance (low value)	-	50 Ω	-
input impedance (high value)	500 kΩ	1 MΩ	-
input frequency range	0.7 µHz	-	50 MHz
input A/D conversion	14 bit, 210 MSample/s		
input noise amplitude (> 10 kHz, AC coupling, 50 Ω and 1 MΩ), for detailed information see Figure 9.5	-	5 nV/√Hz	-
input amplitude accuracy (5 MHz), for detailed information see Figure 9.10	-	-	5%
input amplitude accuracy (10 MHz), for detailed information see Figure 9.10	-	-	10%
input amplitude stability	-	-	0.2 %/°C
input DC offset (<1 V input range)	-	-	20 mV
input DC offset (>1 V input range)	-	-	2%
input bias current	-	-	6 µA
input range settings	1 mV	-	1.5 V
input full range sensitivity (10 V lock-in amplifier output)	1 nV	-	1.5 V
input range (AC) with AC coupling	-0.6V	-	0.6 V
input range (AC) with DC coupling	-1.5 V	-	1.5 V
input range (common mode)	-3.0 V	-	3.0 V
input range (AC + common mode)	-3.3 V	-	3.3 V
dynamic reserve	-	100 dB	120 dB
common mode rejection (CMRR), for detailed information see Figure 9.9	-	75 dB	-

Table 9.6. Reference

Parameter	min	typ	max
internal reference frequency range	0.7 µHz	-	100 MHz
internal reference frequency resolution	0.7 µHz	-	-
internal reference phase range	-180 °	-	180 °
internal reference phase resolution	0.1 µ°	-	-
internal reference acquisition time (lock-time)	instantaneous		
internal reference orthogonality	-	0 °	-
external reference at Input 2/Ref, signal type	arbitrary, active at rising edge		
external reference at Input 2/Ref, frequency range	1 Hz	-	50 MHz

Parameter	min	typ	max
external reference at Input 2/Ref, amplitude - note: for low-swing input signals the gain should be set to full-swing range to achieve best performance	100 mV	-	1 V
external reference at Input 2/Ref, amplitude (using HF2LI-PLL option) - note: for low-swing input signals the gain should be set to full-swing range to achieve best performance	10 mV	-	1 V
external reference at Input 2/Ref, reference acquisition time (lock-time)	-	-	100 reference cycles or 1.2 ms whatever is larger
external reference at DIO0/DIO 1, signal type	digital TTL versus ground		
external reference at DIO0/DIO1, frequency range	1 Hz	-	2 MHz
external reference at DIO0/DIO1, high level	2.0 V	-	5 V
external reference at DIO0/DIO1, low level	0 V	-	0.8 V
external reference at DIO0/DIO1, reference acquisition time (lock-time)	-	-	100 reference cycles or 1.2 ms whatever is larger
external reference at AUXIN1/AUXIN2, signal type	sine or rectangular		
external reference at AUXIN1/AUXIN2, frequency range	1 Hz	-	20 kHz
external reference at AUXIN1/AUXIN2, amplitude	0.5 V	-	1 V
external reference at AUXIN1/AUXIN2, reference acquisition time (lock-time)	-	-	100 reference cycles
auto reference at Input 1/Input 2, signal type	AC signal with zero crossings, AC input setting		
auto reference at Input 1/Input 2, frequency range	1 Hz	-	50 MHz
auto reference at Input 1/Input 2, reference acquisition time (lock-time)	-	-	100 reference cycles or 1.2 ms whatever is larger

Table 9.7. Demodulators

Parameter	Description
demodulator number	HF2IS: 4 dual-phase, 8 dual-phase with multi-frequency kit
	HF2LI: 6
	HF2PLL: 6
demodulator harmonic setting range	1 to 1023

Parameter	Description
demodulator filter time constant	0.8 µs to 580 s
demodulator filter slope / roll-off	6, 12, 18, 24, 30, 36, 42, 48 dB/oct, consisting of up to 8 cascaded critical damping filters
demodulator output resolution	X, Y, R, Θ with 64-bit resolution
demodulator output rate (readout rate), for detailed specifications refer to Table 9.4	on Aux outputs: 921 kS/s
	on USB to host PC: maximum cumulative 700 kS/s
demodulator measurement bandwidth	80 µHz to 200 kHz
demodulator harmonic rejection	max -90 dB
demodulator sinc filter operating range	0.1 Hz to 10 kHz

Table 9.8. HF signal outputs

Parameter	min	typ	max
connectors	front-panel single-ended BNC		
output impedance (Out and Sync)	50 Ω		
input impedance (Add)	1 MΩ		
output frequency range	DC	-	50 MHz
output D/A conversion	16 bit, 210 MSample/s		
output amplitude ranges (restrictions apply for high amplitudes and high frequencies, see Figure 9.11)	±10 mV, ±100 mV, ±1 V, ±10 V		
output maximum current	-	-	100 mA
output amplitude accuracy @ 3 MHz, < 5 V (restrictions apply for high amplitudes and high frequencies, see Figure 9.11)	-	-	1%
output total harmonic distortion THD (1 V, < 10 MHz), see Figure 9.12	-50 dB	-	-
output total harmonic distortion THD (0.1 V, < 10 MHz), see Figure 9.12	-60 dB	-	-
output noise amplitude (frequencies > 10 kHz), 50 Ω termination	-	25 nV/√Hz	-
output phase noise @ 10 MHz, BW = 0.67 Hz, offset 100 Hz	-100 dBc/Hz	-	-
output phase noise @ 10 MHz, BW = 0.67 Hz, offset 1 kHz	-120 dBc/Hz	-	-
output offset amplitude (range setting < 1 V)	-	-	10 mV
output offset amplitude (range setting > 1 V)	-	-	200 mV
input Add signal range	-10 V	-	+10 V
input Add signal bandwidth	DC	-	50 MHz
output Sync signal range (effective range = ±1 * set_amplitude / set_range)	-1 V	-	1 V
output synchronization signal resolution	-	30 µV	-

Table 9.9. Auxiliary Inputs and Outputs

Parameter	Description
auxiliary output connectors	front-panel single-ended BNC
auxiliary output impedance	50 Ω
auxiliary output number and type of signals	4, amplitude, phase, frequency, X/Y, manual
auxiliary output specification	± 10 V, 200 kHz, 16-bit, 921 kSample/s
auxiliary output resolution	0.3 mV
auxiliary input connectors	back-panel single-ended BNC
auxiliary input impedance	1 MΩ
auxiliary input number	2
auxiliary input specification	± 10 V, 100 kHz, 16-bit, 400 kSample/s
auxiliary input resolution	0.3 mV
group delay (lag time from HF input to auxiliary output)	7 µs (typical), 10 µs (maximum)

Table 9.10. Oscillator and clocks

Parameter	min	typ	max
internal oscillator frequency	-	10 MHz	-
internal oscillator output (sine)	-1 V	-	+1 V
internal oscillator initial accuracy	-	-	±30 ppm
internal oscillator aging (stability)	-	-	±5 ppm/year
internal oscillator temperature stability (23° ± 5°)	-	-	±30 ppm
UHS (option) oscillator initial accuracy	-	-	±0.5 ppm
UHS (option) oscillator aging (stability)	-	-	±0.4 ppm/year
UHS (option) oscillator temperature stability (23° ± 5°)	-	-	±0.03 ppm
UHS (option) oscillator phase noise (at 100 Hz)	-130 dBc/Hz	-	-
UHS (option) oscillator phase noise (at 1 kHz)	-140 dBc/Hz	-	-
UHS (option) oscillator reference stability (over 30 s)	0.00005 ppm	-	-
UHS (option) oscillator time to reach specification	-	-	60 s
external clock connector	back-panel single-ended BNC		
external clock input impedance	1 MΩ		
external clock input voltage	0 V	-	+3.3 V
external clock frequency	9.98 MHz	10 MHz	10.02 MHz

9.3. Digital Interface Specifications

Table 9.11. Digital interfaces

Parameter	Description
host computer connection	USB 2.0 high-speed, 480 Mbit/s
ZCtrl pre-amplifier control bus	proprietary bus to control external pre-amplifiers
ZSync synchronization bus	proprietary bus to locally interconnect ZI instruments
DIO connector	32 bit, general purpose

The DIO connector is a HD 68 pin connector, typically also used by SCSI-2 and SCSI-3 interfaces, 47 mm wide male connector. The DIO port features 16 bits that can be configured byte-wise as inputs or outputs, as well as 16 input only bits. The digital signals follow the CMOS/TTL specification.

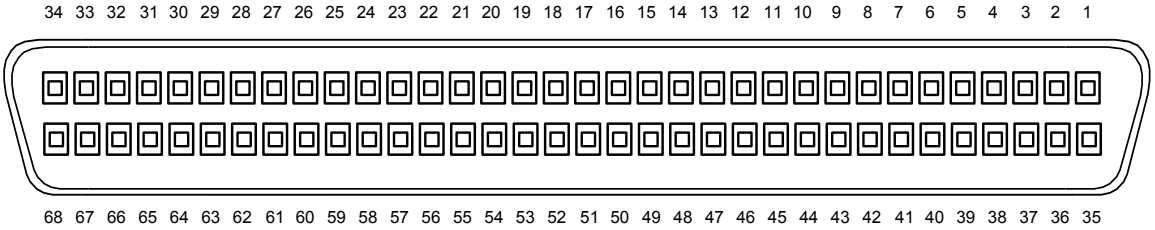


Figure 9.1. DIO HD 68 pin connector

Table 9.12. DIO pin assignment

Pin	Name	Description	Range specification
68	CLKI	clock input, used to latch signals at the digital input ports - can also be used to retrieve digital signals from the output port using an external sampling clock	5 V CMOS/TTL
67	DOL	DIO output latch, 64 MHz clock signal, the digital outputs are synchronized to the falling edge of this signal	5 V CMOS
66-51	DI[31:16]	digital input	digital input CMOS/TTL level
50-35	DIO[15:0]	digital input or output (set by user)	output CMOS 5 V, input is CMOS/TTL
34-3	GND	digital ground	-
2-1	PWR	5 V supply (100 mA max)	-

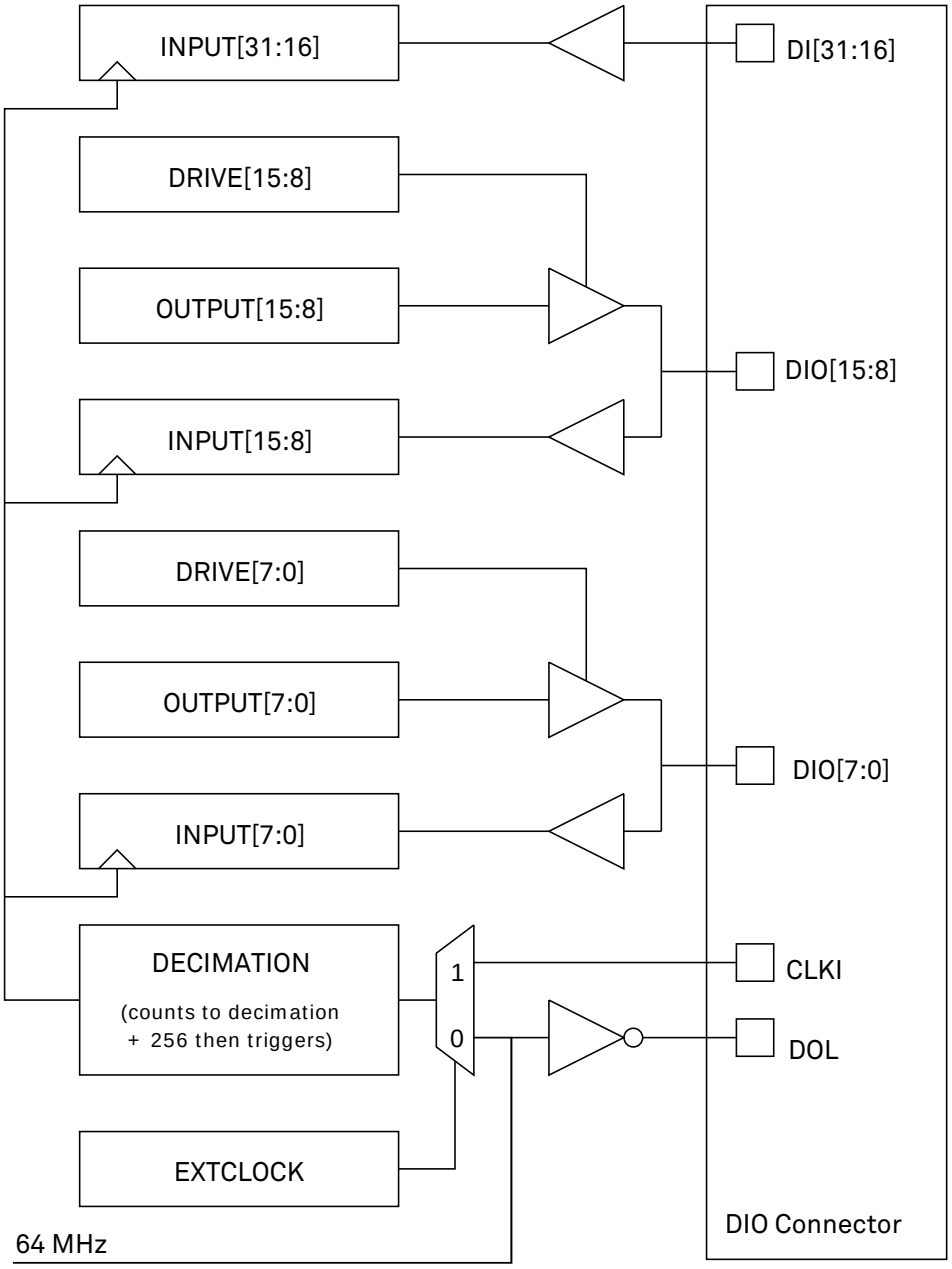


Figure 9.2. DIO input/output architecture

The HF2 Digital I/O Breakout Board provides an easy way to access all pins of the DIO Connector. The board consists of 68 pin headers and a 68-pin female socket to be connected to the HF2 using a ribbon cable. For description of the pins, refer to the Table 9.12 . The HF2 DIO Breakout Board is available with Zurich Instruments on demand.

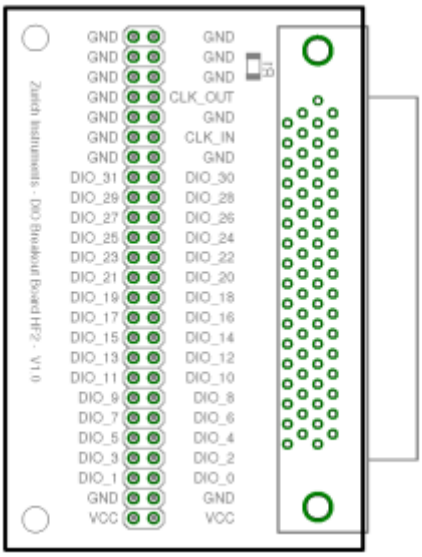


Figure 9.3. HF2 digital I/O breakout board

The internally generated 10 MHz clock is made available for external synchronization at the ZSync Out RJ45 connector. The clock signal is at pin 1, ground at pin 2. To connect: simply prepare a cable assembly that allows you to connect the 10 MHz signal from the HF2 to the BNC input of the other device external clock.

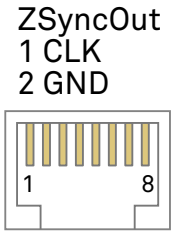
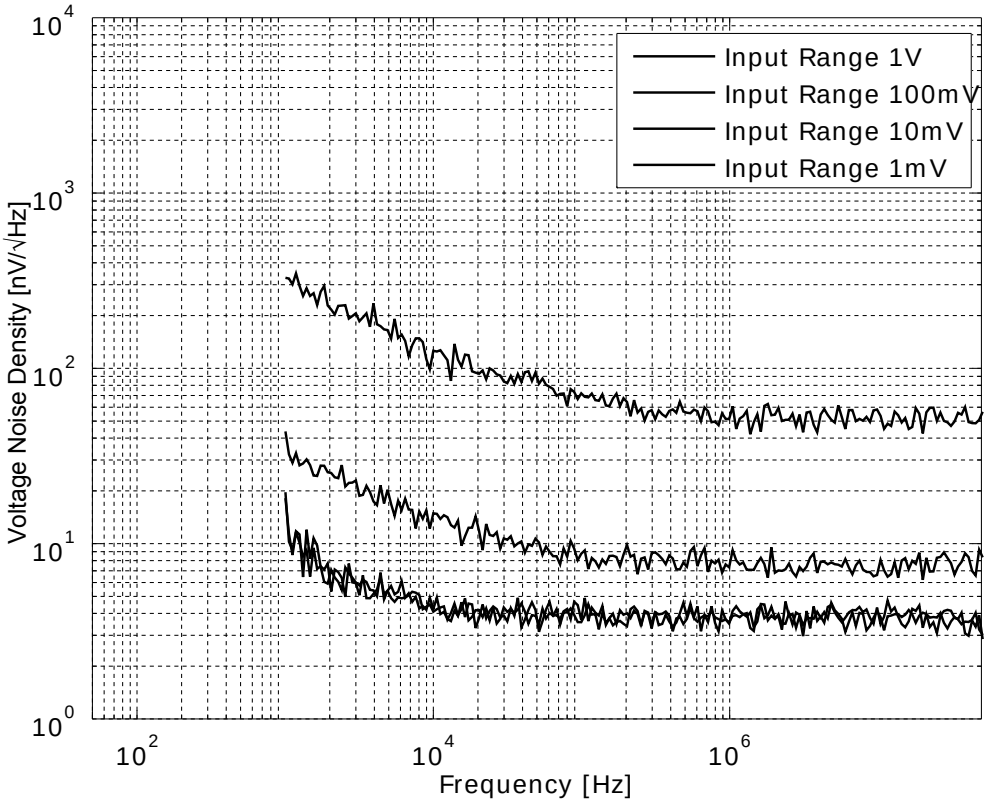


Figure 9.4. The pinout of the RJ45 jack

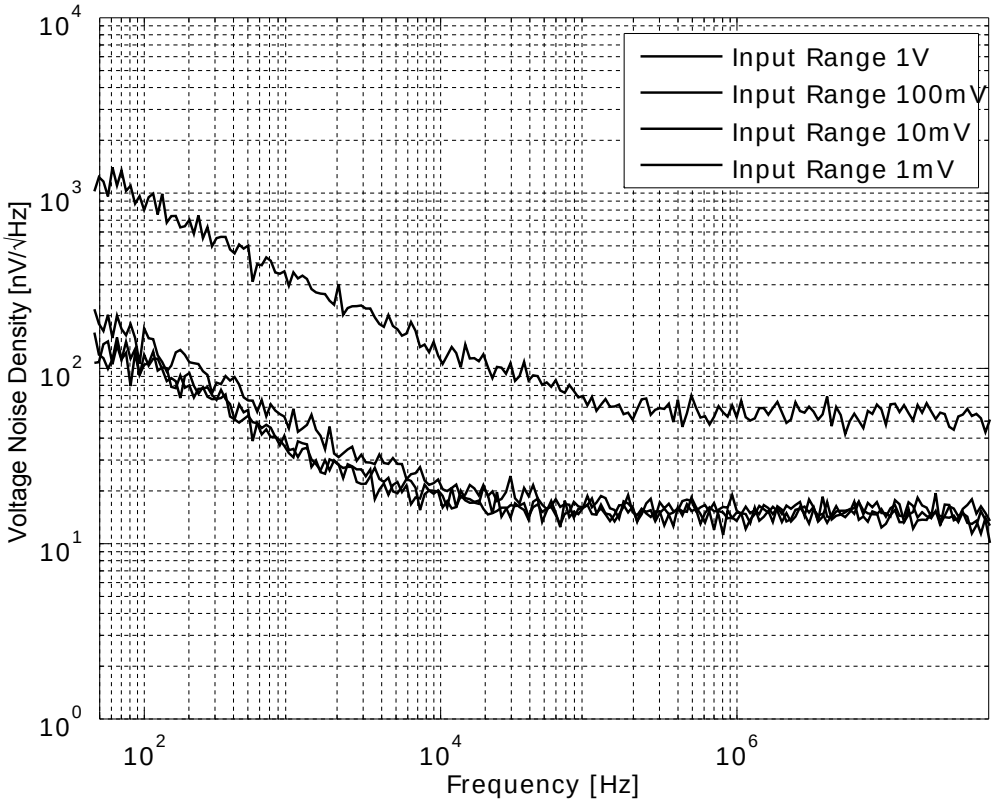
9.4. Performance Diagrams

Many parameters mentioned in Section 9.2 are valid without specific conditions. Other parameters instead are typical specifications because they depend on several parameters, such as range settings, and frequency. This section completes the previous chapters with detailed performance diagrams in order to support the validation of applications.



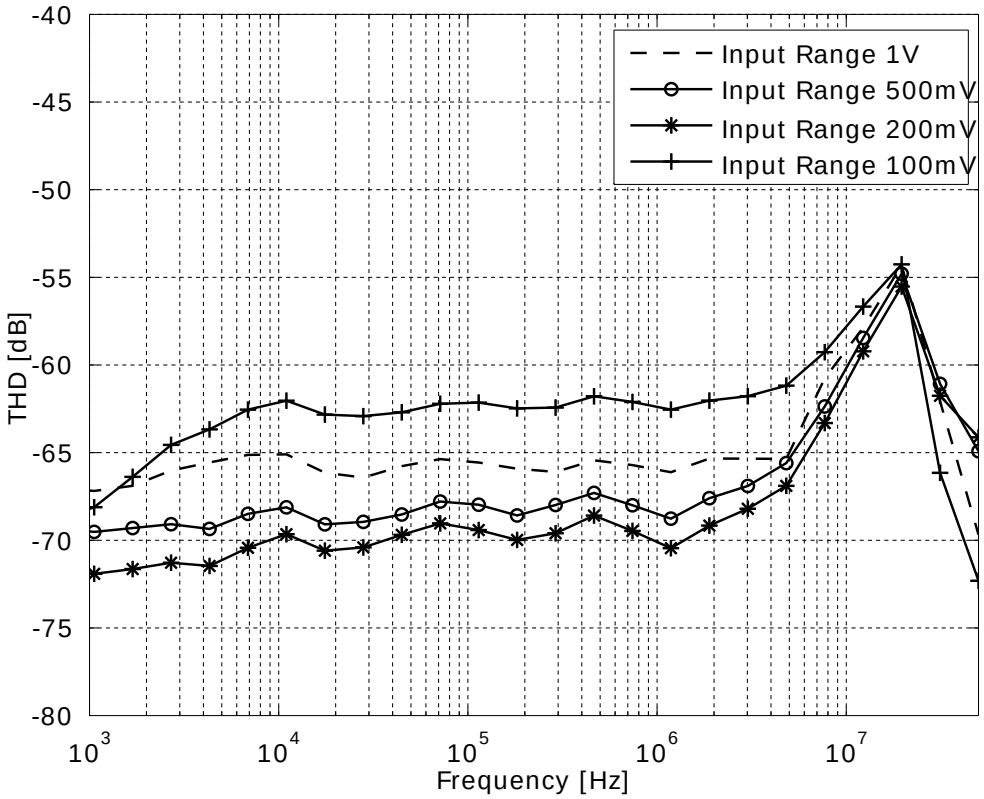
Input noise amplitude depends on several parameters, and in particular on the frequency and the setting for the input range. The noise is lower for smaller input ranges, and it is recommended to perform noise measurements with the AC coupling setting. In AC coupling mode, both 10 mV and 1 mV signal ranges have the same input noise performance. The corner frequency of the 1/f noise is in the range of 10 kHz and the white noise floor is around 5 nV/√Hz in AC coupling mode.

Figure 9.5. HF input noise with AC coupling



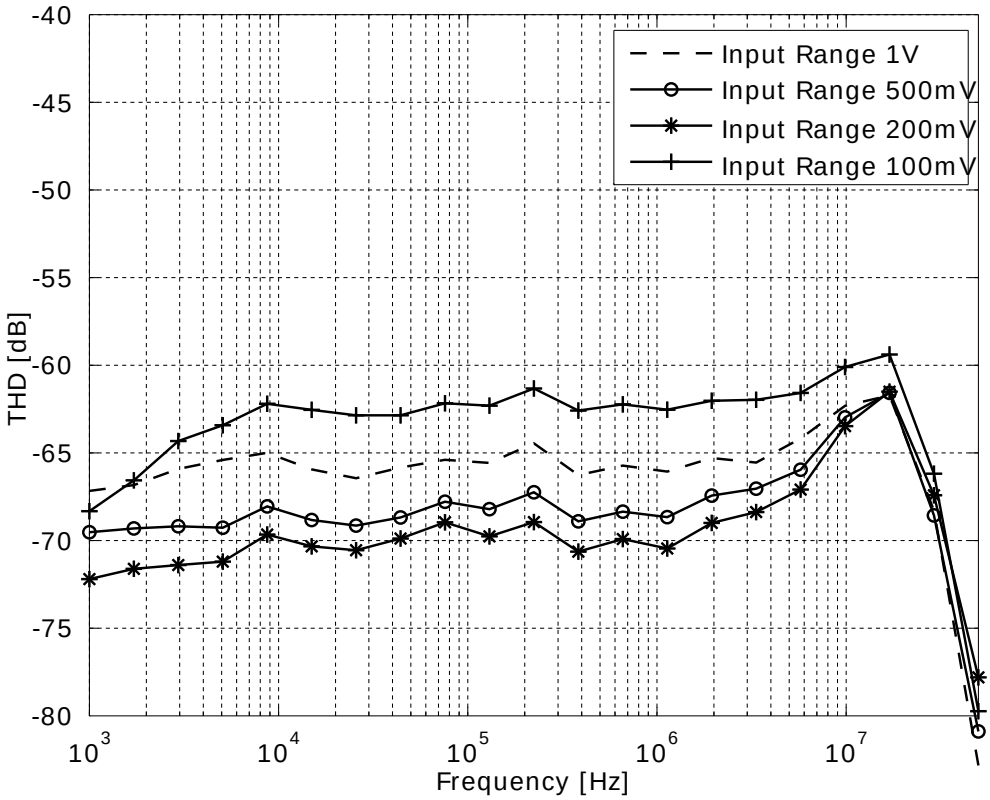
Input noise amplitude depends on several parameters, and in particular on the frequency and the setting for the input range. The noise is lower for smaller input ranges. The corner frequency for AC coupling is at 1 kHz. Therefore, for noise measurements above 1 kHz, AC coupling should be used, for noise measurements below 1 kHz, DC coupling should be used. In DC coupling mode, the input noise for low ranges in limited by the coupling selection. The corner frequency of the 1/f noise is in the range of 10 kHz and the white noise floor is around 15 nV/√Hz in DC coupling mode.

Figure 9.6. HF input noise with DC coupling



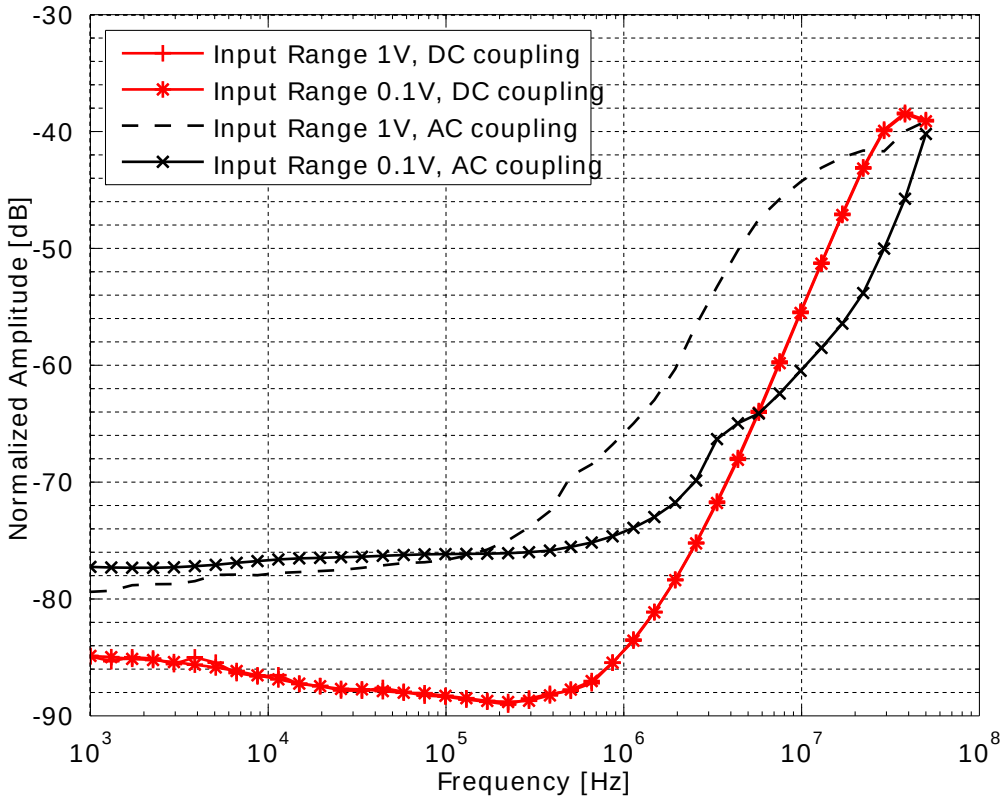
Input total harmonic distortion (THD) depends on several parameters, and in particular on the frequency and the setting for the input range. The test is performed with the input amplitude at 50% of the range setting. For frequencies below 5 MHz input THD is below -60 dB for any range setting. For frequencies above 10 MHz the AC coupling mode inserts about 5 dB more distortion than the DC mode. The total harmonic distortion is calculated from the measurement of the second and the third harmonic.

Figure 9.7. HF input total harmonic distortion with AC coupling



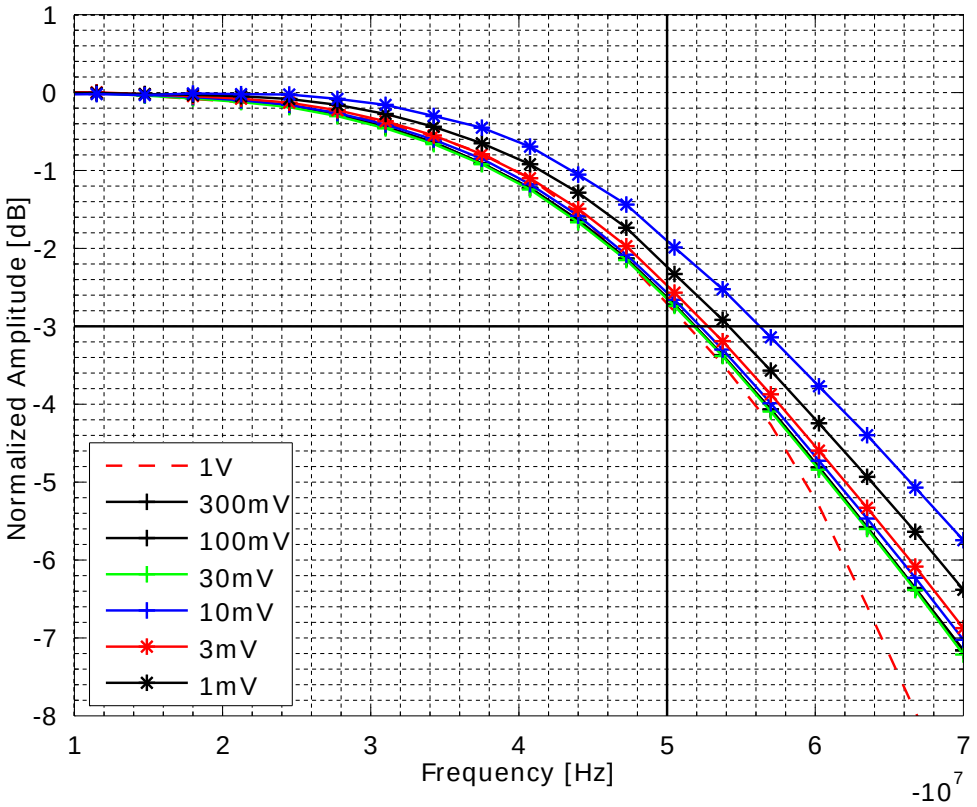
Input total harmonic distortion (THD) depends on several parameters, and in particular on the frequency and the setting for the input range. The test is performed with the input amplitude at 50% of the range setting. For frequencies below 10 MHz input THD is below -60 dB for any range setting. Smaller range settings can be as good as -70 dB. The total harmonic distortion is calculated from the measurement of the second and the third harmonic.

Figure 9.8. HF input total harmonic distortion with DC coupling



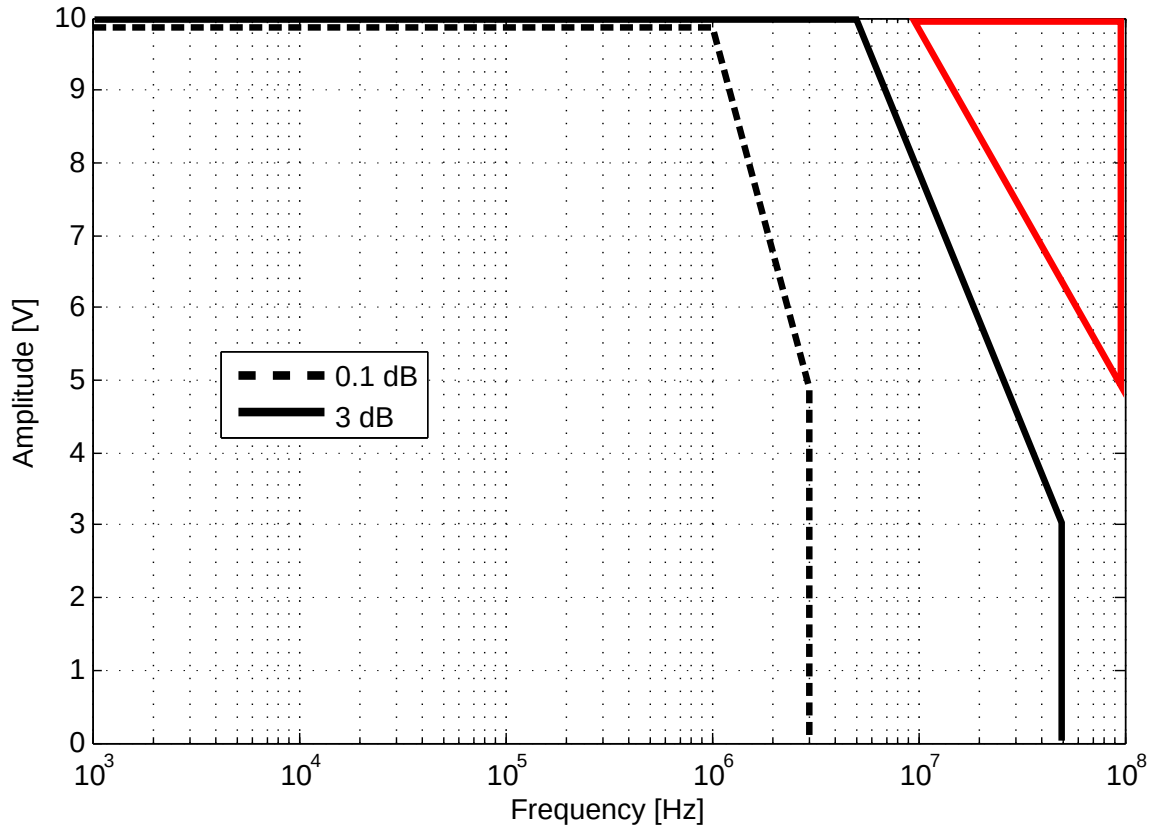
Input common mode rejection ratio (CMMR) of the signal input for different frequencies and different input amplitudes. For this test the same 2 V_{pp} signal is applied to the differential inputs on the instrument and the differential measurement is captured. The CMMR is better in DC mode for frequencies up to 7 MHz. Above 7 MHz, the CMMR in AC mode is better suited, but only for amplitudes up to 100 mV.

Figure 9.9. HF input common mode rejection ratio



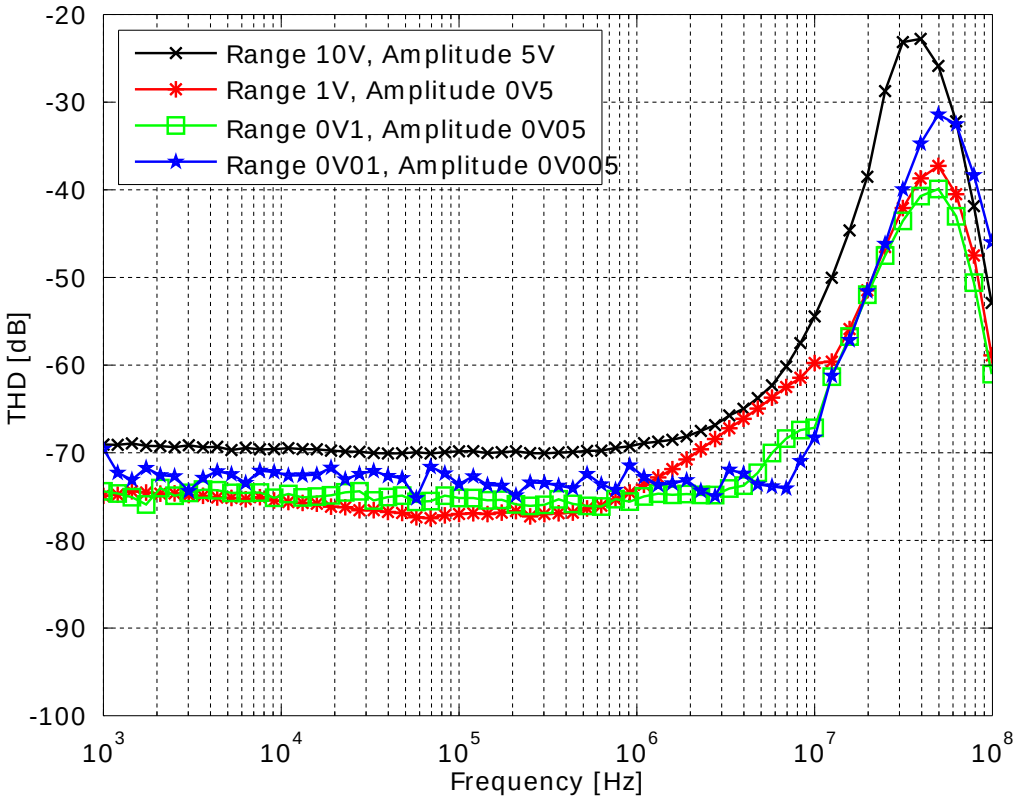
Signal input bandwidth versus frequency for different range settings. The vertical solid black line marks the 50 MHz specification and the horizontal solid black line corresponds to 3 dB attenuation. The bandwidth is between 50 MHz and 60 MHz for all displayed range settings (intermediate range selections can be chose in the graphical user interface).

Figure 9.10. HF input bandwidth



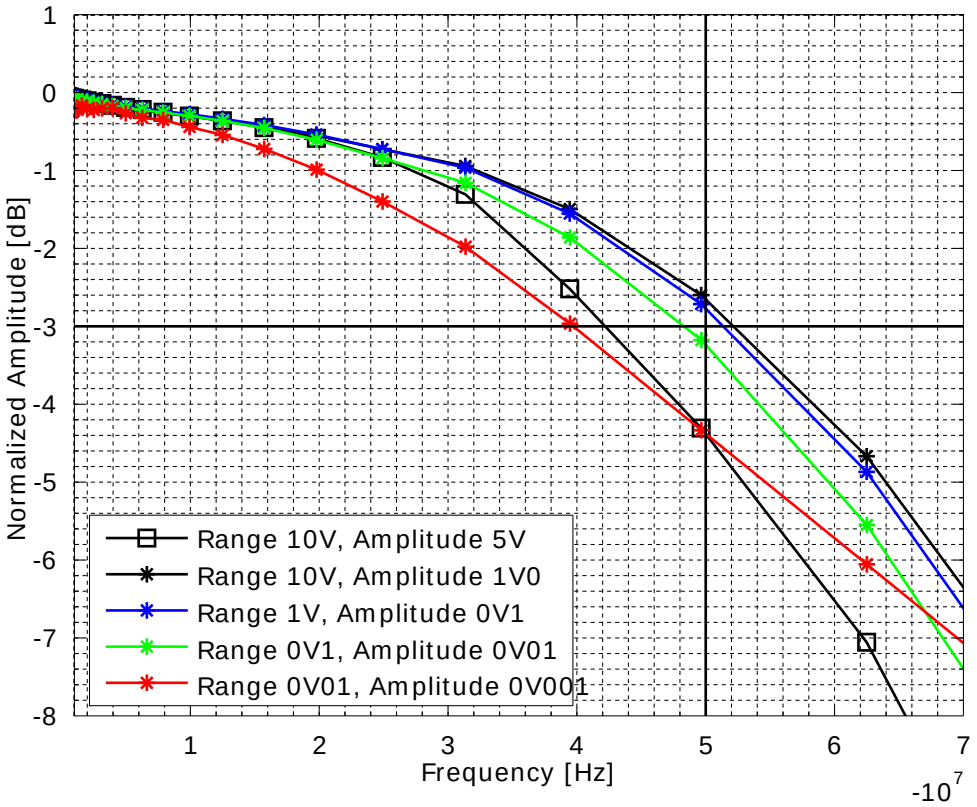
The amplitude accuracy of the output signal generator depends on the frequency and the amplitude. The plot shows the output amplitude accuracy region for 0.1dB (roughly 1%) and 3 dB (around 40%) of the expected value. The measurement was performed with 50 Ω load and is better for larger loads. The plot shows that the signal amplitude up to 1 MHz is accurate at 0.1 dB at any output voltage (area within dashed line). Above this frequency, the anti-aliasing filters at 50 MHz impact the absolute value of the generated signal: the output amplitude is reduced. Above 5 MHz the maximum amplitude is below 10 V, and decreases further reaching 3 V_{RMS} at 50 MHz (area within solid line). It is not recommended to operate the instrument at high amplitudes (above 5 V_{RMS}) and high frequencies (above 10 MHz) because of the large signal distortion and potential longterm damage to the output drivers (area within solid red line).

Figure 9.11. HF output amplitude accuracy



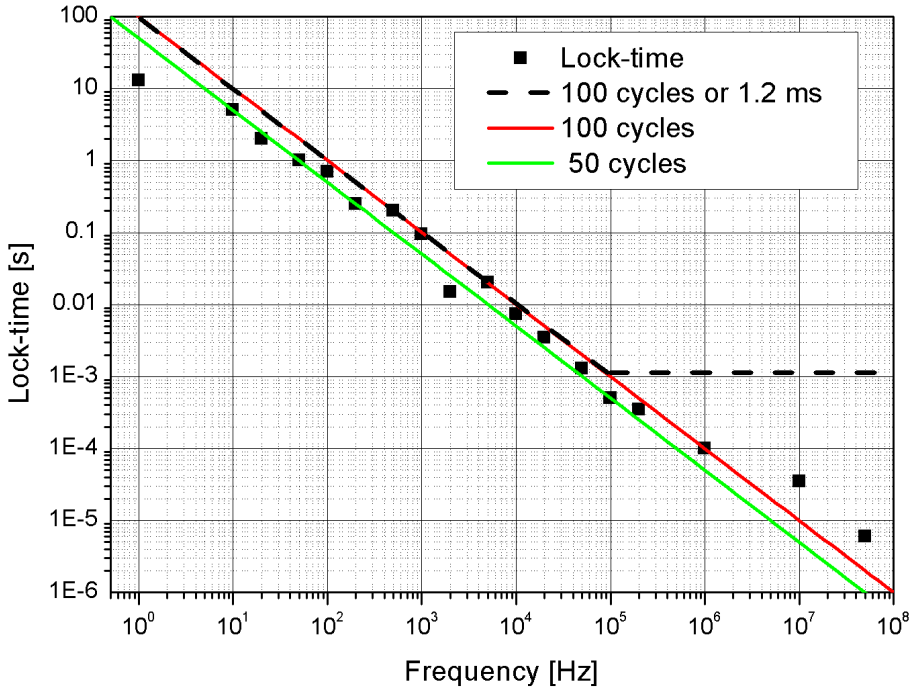
The total harmonic distortion of the signal outputs was measured with 50 Ω load and by applying amplitude at 50% of the corresponding range. The THD is around 75 dB for signals below 0.5 V, and increases for larger amplitudes. The distortion increases significantly at frequencies above 10 MHz. The total harmonic distortion is calculated from the measurement of the second and the third harmonic.

Figure 9.12. HF output total harmonic distortion



The vertical solid black line marks the 50 MHz specification and the horizontal solid black line corresponds to 3 dB attenuation. The bandwidth depends on the range and amplitude. The measurement is performed with 50 Ω load.

Figure 9.13. HF output bandwidth



The dashed black line marks the specified lock-time corresponding to 100 reference cycles or 1.2 ms. The 100 and 50 cycle are also plotted for convenience. The graph indicates that the lock-time of the HF2 Instruments follows the 100 reference cycles spec, is slightly worse for frequencies above 1 MHz, and is better for frequency below 1 Hz.

Figure 9.14. Lock-time

9.5. Ground and Earth Scheme

Ground loops have effect to introduce noise at the line frequency (mostly 50/60 Hz) and higher harmonics, and aliasing in the demodulated signal that is measurable for frequencies up to 10 MHz. Some lock-in amplifiers implement a line filter which has the effect to exclude low frequency measurements. This is not the case for the HF2 Instruments where an effective ground strategy is implemented.

In order to suppress large signal components at line frequency and higher harmonics avoiding ground loops within the measurement setup is required. Possible reasons for line frequency components include parasitics resistances between the different signal grounds, inductive coupling from line transformers and other electrical apparatus into the signal paths, and pre-amplifiers that generate additional loops.

Counter measures are to break loops, using differential wiring, implementing star ground connections in the measurement setup, with the main ground closest to the setup as possible, connect all instrument casing to earth, and using optocouplers and transformers where a galvanic decoupling is provided in the signal path.

The grounding of the HF2 Instrument is implemented connecting analog ground and digital ground in a star network. This reduces the digital ground noise that flows into the analog domain considerably. All analog grounds are connected together before they are connected to the digital ground (e.g. USB ground). All grounds are decoupled by the Earth by means of a 1 MΩ resistor, which is however generally shorted by a PC connected by means of a USB cable. The earth connection of the power plug connects at the same time the chassis and the banana plug on the rear Instrument panel.

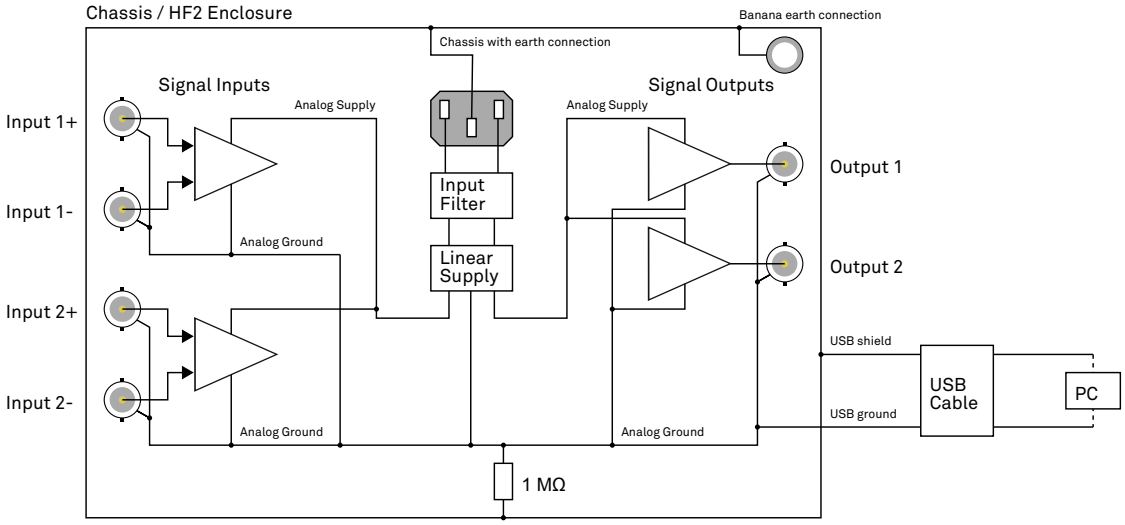


Figure 9.15. Instrument ground & earth connection scheme

Some applications however require floating ground applications, it is suggested to make use of the differential inputs, by connecting the BNC shield to the negative BNC connector. The limitation for this strategy, is that the floating ground should not exceed the specified maximum input common mode offset.

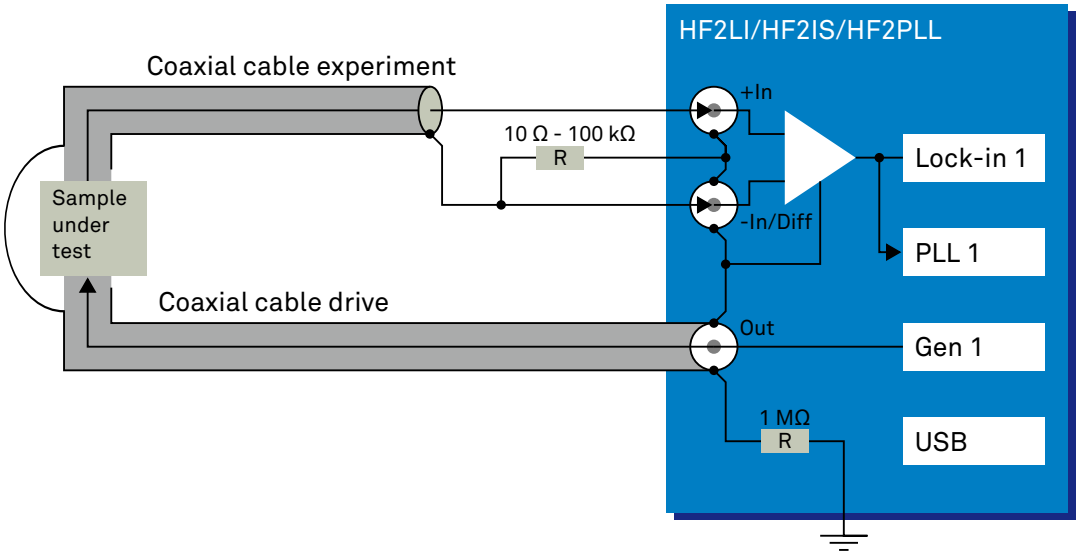
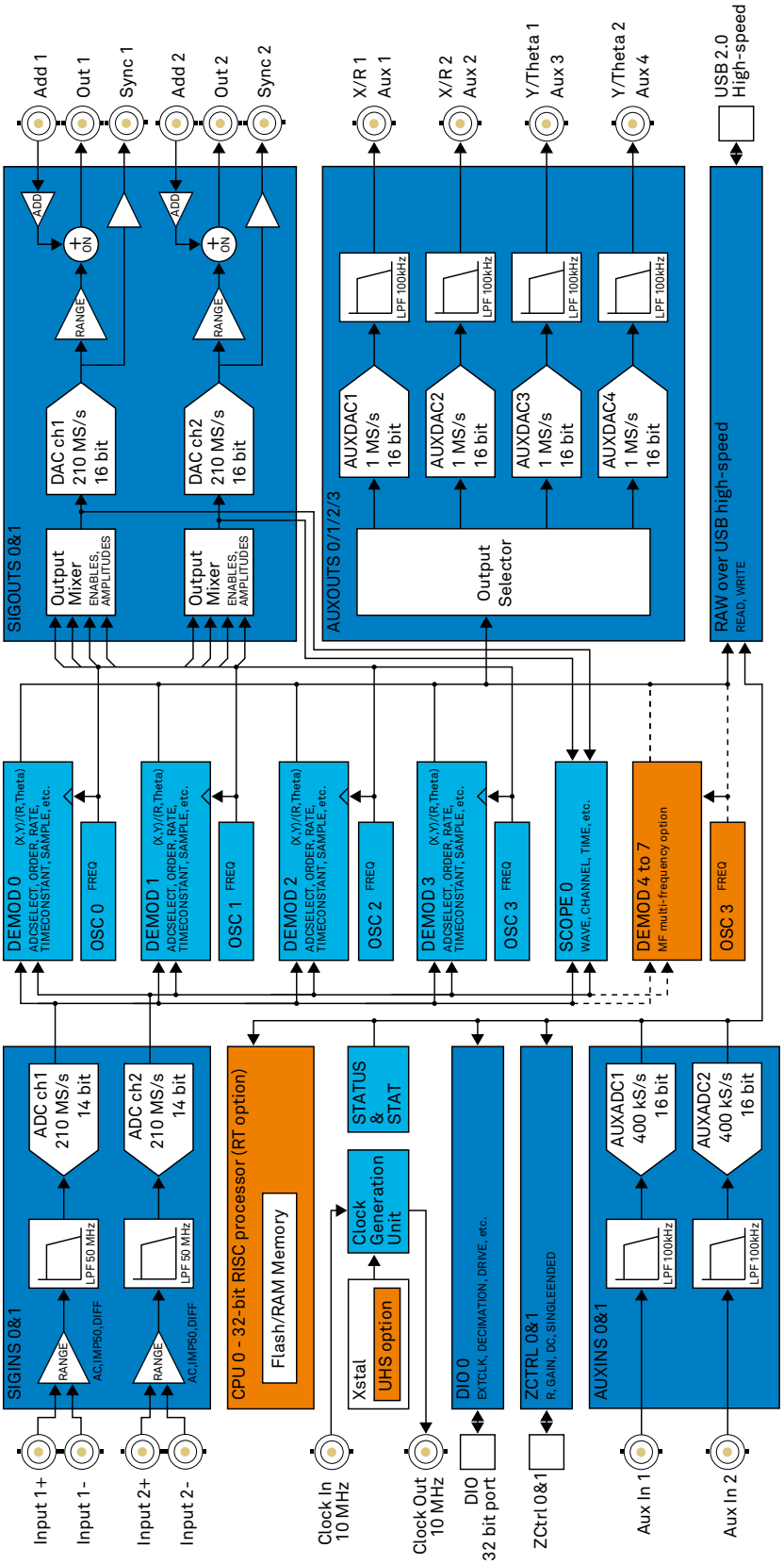
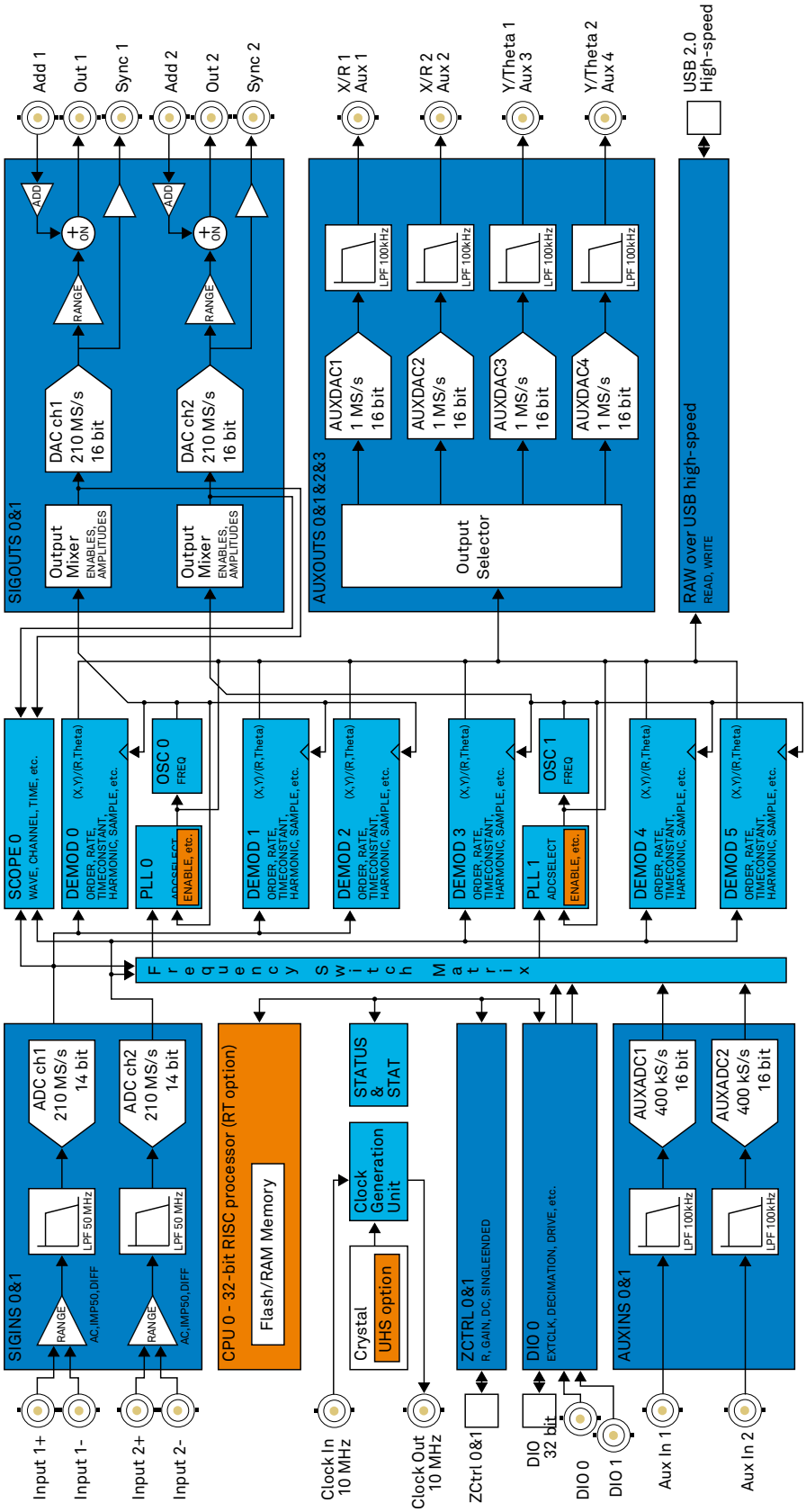


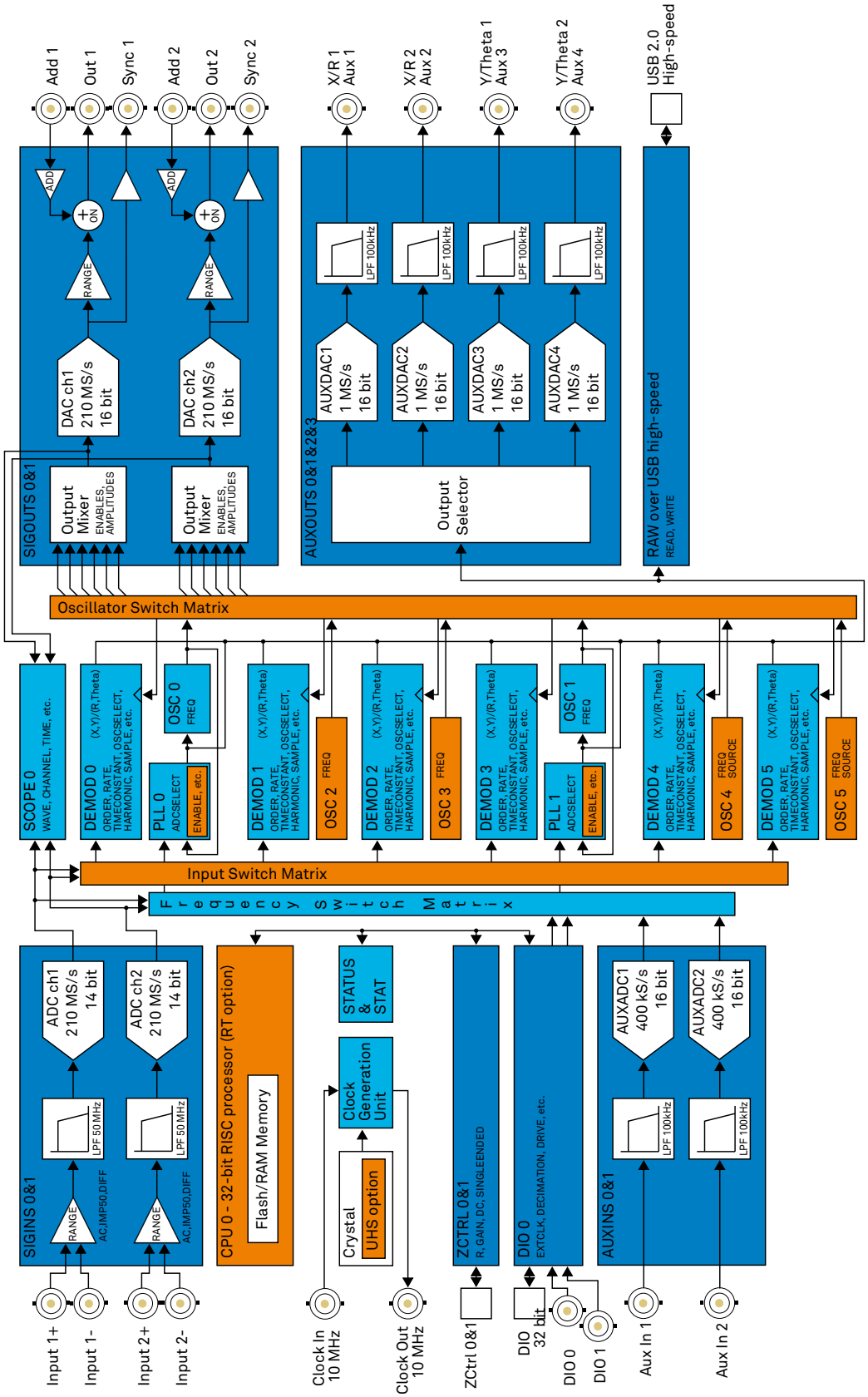
Figure 9.16. Differential connection scheme reducing ground loops

9.6. Reference Images

The following figures are intended for advanced users with programming projects on the HF2 Instruments.


Add 1
Out 1
Sync 1
Add 2
Out 2
Sync 2
X/R 1
Aux 1
X/R 2
Aux 2
Y/Theta 1
Aux 3
Y/Theta 2
Aux 4
USB 2.0
High-speed
SIGOUTS 0&1
ADD
ON
RANGE
DAC ch1
210 MS/s
16 bit
DAC ch2
210 MS/s
16 bit
Output
Mixer
ENABLES,
AMPLITUDES
AUXOUTS 0/1/2/3
LPF 100kHz
AUXDAC1
1 MS/s
16 bit
AUXDAC2
1 MS/s
16 bit
AUXDAC3
1 MS/s
16 bit
AUXDAC4
1 MS/s
16 bit
Output
Selector
RAW over USB high-speed
READ, WRITE
DEMOD 0
DEMOD 1
DEMOD 2
DEMOD 3
(X,Y)/(R,Theta)
ADCSELECT, ORDER, RATE,
TIMECONSTANT, SAMPLE, etc.
OSC 0
OSC 1
OSC 2
OSC 3
FREQ
SCOPE 0
WAVE, CHANNEL, TIME, etc.
DEMOD 4 to 7
MF multi-frequency option
SIGINS 0&1
LPF 50 MHz
AC,IMP50,DIFF
ADC ch1
210 MS/s
14 bit
ADC ch2
210 MS/s
14 bit
CPU 0 - 32-bit RISC processor (RT option)
Flash/RAM Memory
STATUS
&
STAT
Clock
Generation
Unit
Xstal
UHS option
DIO 0
EXTCLK, DECIMATION, DRIVE, etc.
ZCTRL 0&1
R, GAIN, DC, SINGLEENDED
AUXINS 0&1
AUXADC1
400 kS/s
16 bit
AUXADC2
400 kS/s
16 bit
Input 1+
Input 1-
Input 2+
Input 2-
Clock In
10 MHz
Clock Out
10 MHz
DIO
32 bit port
ZCtrl 0&1
Aux In 1
Aux In 2


Add 1
Out 1
Sync 1
Add 2
Out 2
Sync 2
X/R 1 Aux 1
X/R 2 Aux 2
Y/Theta 1 Aux 3
Y/Theta 2 Aux 4
USB 2.0 High-speed
SIGOUTS 0&1
ADD
ON
RANGE
DAC ch1 210 MS/s 16 bit
DAC ch2 210 MS/s 16 bit
Output Mixer ENABLES, AMPLITUDES
AUXOUTS 0&1&2&3
LPF 100kHz
AUXDAC1 1 MS/s 16 bit
AUXDAC2 1 MS/s 16 bit
AUXDAC3 1 MS/s 16 bit
AUXDAC4 1 MS/s 16 bit
Output Selector
RAW over USB high-speed
READ, WRITE
SCOPE 0
WAVE, CHANNEL, TIME, etc.
DEMOD 0
DEMOD 1
DEMOD 2
DEMOD 3
DEMOD 4
DEMOD 5
(X,Y)/(R,Theta)
ORDER, RATE, TIMECONSTANT, HARMONIC, SAMPLE, etc.
OSC 0
OSC 1
FREQ
PLL 0
PLL 1
ADCSELECT
ENABLE, etc.
Frequency Switch Matrix
SIGINS 0&1
LPF 50 MHz
AC,IMP50,DIFF
ADC ch1 210 MS/s 14 bit
ADC ch2 210 MS/s 14 bit
CPU 0 - 32-bit RISC processor (RT option)
Flash/RAM Memory
Crystal
UHS option
Clock Generation Unit
STATUS & STAT
ZCTRL 0&1
R, GAIN, DC, SINGLEENDED
DIO 0
EXTCLK, DECIMATION, DRIVE, etc.
AUXINS 0&1
AUXADC1 400 kS/s 16 bit
AUXADC2 400 kS/s 16 bit
Input 1+
Input 1-
Input 2+
Input 2-
Clock In 10 MHz
Clock Out 10 MHz
ZCtrl 0&1
DIO 32 bit
DIO 0
DIO 1
Aux In 1
Aux In 2


Add 1
Out 1
Sync 1
Add 2
Out 2
Sync 2
X/R 1 Aux 1
X/R 2 Aux 2
Y/Theta 1 Aux 3
Y/Theta 2 Aux 4
USB 2.0 High-speed
SIGOUTS 0&1
DAC ch1 210 MS/s 16 bit
DAC ch2 210 MS/s 16 bit
Output Mixer ENABLES, AMPLITUDES
AUXOUTS 0&1&2&3
AUXDAC1 1 MS/s 16 bit
AUXDAC2 1 MS/s 16 bit
AUXDAC3 1 MS/s 16 bit
AUXDAC4 1 MS/s 16 bit
Output Selector
RAW over USB high-speed
READ, WRITE
Oscillator Switch Matrix
SCOPE 0
DEMOD 0
DEMOD 1
DEMOD 2
DEMOD 3
DEMOD 4
DEMOD 5
PLL 0
PLL 1
OSC 0
OSC 1
OSC 2
OSC 3
OSC 4
OSC 5
Input Switch Matrix
Frequency Switch Matrix
SIGINS 0&1
ADC ch1 210 MS/s 14 bit
ADC ch2 210 MS/s 14 bit
CPU 0 - 32-bit RISC processor (RT option)
Flash/RAM Memory
STATUS & STAT
Clock Generation Unit
Crystal
UHS option
ZCTRL 0&1
DIO 0
AUXINS 0&1
AUXADC1 400 kS/s 16 bit
AUXADC2 400 kS/s 16 bit
Input 1+
Input 1-
Input 2+
Input 2-
Clock In 10 MHz
Clock Out 10 MHz
ZCtrl 0&1
DIO 32 bit
DIO 0
DIO 1
Aux In 1
Aux In 2

9.7. Test Specifications

This chapter describes the performance tests for the Zurich Instruments HF2 Instruments. Users are encouraged to verify that the instrument performs as specified, not only after shipping but also to ensure continuous performance over time. Some of the measurements described are setup as a self-test, where no additional measurement equipment is needed. A few tests, however, require additional measurement equipment, such as the references needed for the accuracy tests.

The HF2 instruments have two analog input/output channels, four auxiliary output and two auxiliary analog input channels. Tests described here are performed on one channel of each type and need to be repeated accordingly on the other channels.

The following table gives a list of equipment that meets the accuracy requirements for the described tests.

Table 9.13. Required equipment

Equipment	Specifications	Recommendation
HF2 base instrument	No options required	HF2LI, HF2IS, HF2PLL
Function generator	>50 MHz range, sine waveform generation	Agilent 33250A
Digital multimeter	0.1 mV resolution, 20 V range	Agilent 34410A
BNC cables	2 x 50 Ω , male-to-male cables (<50cm)	
BNC T-connector	1 x 50 Ω, male-female-female connector	

In addition, the following need to be fulfilled:

- The test equipment should be connected to a common AC power circuit. Connecting the test equipment to separate AC power circuits can result in offset voltages between them and will introduce an additional error into the testing procedure. If the user is unsure about the AC power circuit layout, a common power strip for all the test equipment is advised.
- Allow the test equipment to warm up for at least 30 minutes.
- Operating temperature and humidity should be within the instrument's specified range.
- Make sure that the latest versions of the LabOne and ziControl software packages have been installed on the host computer. More information about installing software can be found in the Getting Started Chapter .

During testing, the user should adopt the measurement setup and instrument settings as described for each specific test. The user should compare the obtained tests results with the HF2 instrument's specifications. HF2 connection to the host computer via USB 2.0 cable is assumed and indicated in all the described setups, unless stated otherwise.

9.7.1. Test Input Noise

Definition

Input noise voltage (INV) is the sum of all the noise sources internal to the instrument, referenced to the signal input. Noise is expressed in its power spectral density (noise power per unit frequency) in units V^2/Hz, and more typically, its square root in units $V/\sqrt{Hz}$.

Typical HF2 Instrument settings that influence the noise at the input are input range, input impedance, measurement frequency and the selected input coupling (AC or DC).

Setup

Disconnect all BNC cables from the HF2's inputs and outputs. The input noise test setup is shown in the figure below.

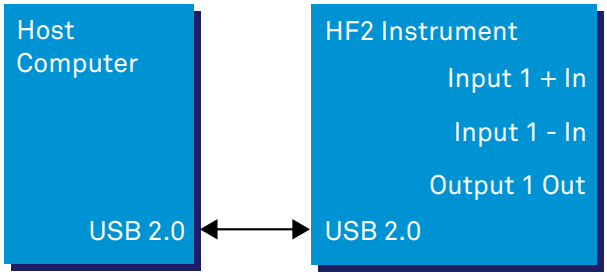


Figure 9.20. Input noise test setup

The HF2 instrument settings on the ziControl panel should be according to the table below:

Table 9.14. HF2 instrument settings

Ch1 Signal Input Range	1 mV
Ch1 Signal Input AC/ Diff/ 50	ON/ OFF/ ON
Ch1 Filter dB/Oct	24
Ch1 Filter BW	10 Hz
Ch1 Demodulator 1 Readout	Cont.
Ch1 Frequency (Hz)	Variable

Measurement

The measurements are performed using the frequency sweeper with the following settings:

Table 9.15. Frequency sweeper settings

Sweep Range Start	1 kHz
Sweep Range Stop	50 MHz
Sweep Range Points	100
Sweep Range Log Sweep	ON
Auto BW	ON
Settling Time	10 TC
Averaging	32

Result Unit	$V_{RMS}/\sqrt{Hz}$
Display Polar	ON

After the settings have been applied, start the measurement by pressing the Sweep Control Single button. The plots of the input noise (units of $V_{RMS}/\sqrt{Hz}$), as a function of frequency, in the range from 1 kHz to 50 MHz are displayed in the frequency sweeper. The figures below show the results obtained with AC and DC coupling.

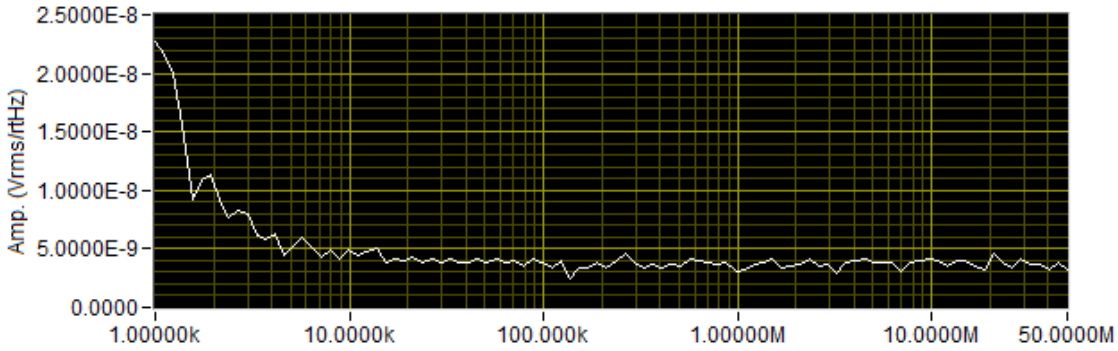


Figure 9.21. Input noise test results with AC coupling

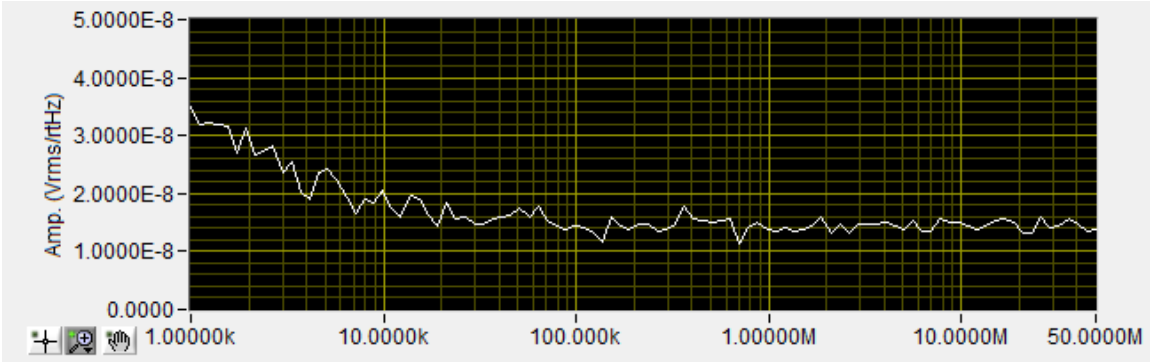


Figure 9.22. Input noise test results with DC coupling

The noise spectral density of the input channel with AC coupling is below 5 $nV_{RMS}/\sqrt{Hz}$ in the frequency range from 10 kHz to 50 MHz. In the same frequency range and with the DC coupling, the input noise is around 15 $nV_{RMS}/\sqrt{Hz}$. The noise spectral density at low frequencies (below 10 kHz) is dominated by $1/f$ noise. Users who are interested in the noise spectral density below 1 kHz should perform their tests using the DC coupling and longer time constants, which will result in longer measurement time.

9.7.2. Test Dynamic Reserve

Definition

The dynamic reserve (DR) of a lock-in amplifier is a measure of its capability to withstand the disturbing signals and noise at non-reference frequencies, while maintaining the specified measurement accuracy within the signal bandwidth. It is usually given in units of dB and expresses the ratio between the amplitude of the disturbing signal ($V_{disturb}$) and the minimal signal amplitude that is measured with 1% accuracy (V_{meas}) and can be calculated using the formula,

$$DR = 20 * \log\left(\frac{V_{disturb}}{V_{meas}}\right) \quad (9.1)$$

Typical HF2 Instrument settings that influence the dynamic reserve are the measurement and the disturbing signal frequency, the input range, the demodulator filter settings and the selected input coupling (AC or DC).

Setup

Connect Signal Output 1 Out and the Signal Input 1 + In using a short BNC cable. Use second BNC cable to connect the Signal Output 2 Out and the Signal Input 1 - In Diff. The Output 1 provides V_{meas}, and the Output 2 $V_{disturb}$ signal. The setup is shown in the figure below.

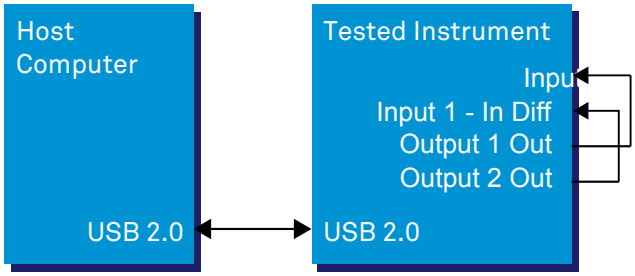


Figure 9.23. Dynamic reserve test setup

The HF2 instrument settings on the ziControl panel should be according to the table below:

Table 9.16. HF2 instrument settings

Ch1 Signal Inputs Range	1.2 V
Ch1 Signal Inputs AC/ Diff/ 50	ON/ ON/ OFF
Ch1 Frequency	Variable
Ch1 Filters dB/Oct	24/48
Ch1 Filters BW	Variable
Ch1 Demodulators 1	ON
Ch1 Outputs range	10 mV
Ch1 Outputs Amplitude	6.943 µV
Ch2 Outputs range	1.0 V
Ch2 Outputs Amplitude	1.0 V
Ch2 Frequency (Hz)	1.1 MHz

Measurement

Two test methods are explained in this section. The first method uses the frequency sweeper with the following settings:

Table 9.17. Frequency sweeper settings

Sweep Range Start	1.095 MHz
Sweep Range Stop	1.105 MHz
Sweep Range Points	101
Auto BW	OFF
Settling Time	10 TC eff
Filter Averaging	32
Display Coord. Sys.	Polar
Mapping	dB
Result Unit	Vpk
Persistent Display	Manual Memorize
Display Depth #	2

The Sweeper tool directly measures the dynamic reserve as a function of frequency. The measurement is done with $V_{disturb}$= 1 V, at a fixed frequency of 1.1 MHz, while the V_{meas} = 6.943 µV signal's frequency is swept in the frequency window of 10 kHz around 1.1 MHz. One may perform two measurements by setting the demodulators filter bandwidth BW 3dB, first to 10 Hz, then to 1 Hz. Start each measurement by pressing the Sweep Control Single button in order to plot the amplitude and phase graphs. The white curve is taken with bandwidth of 10 Hz and the red one with bandwidth of 1 Hz.

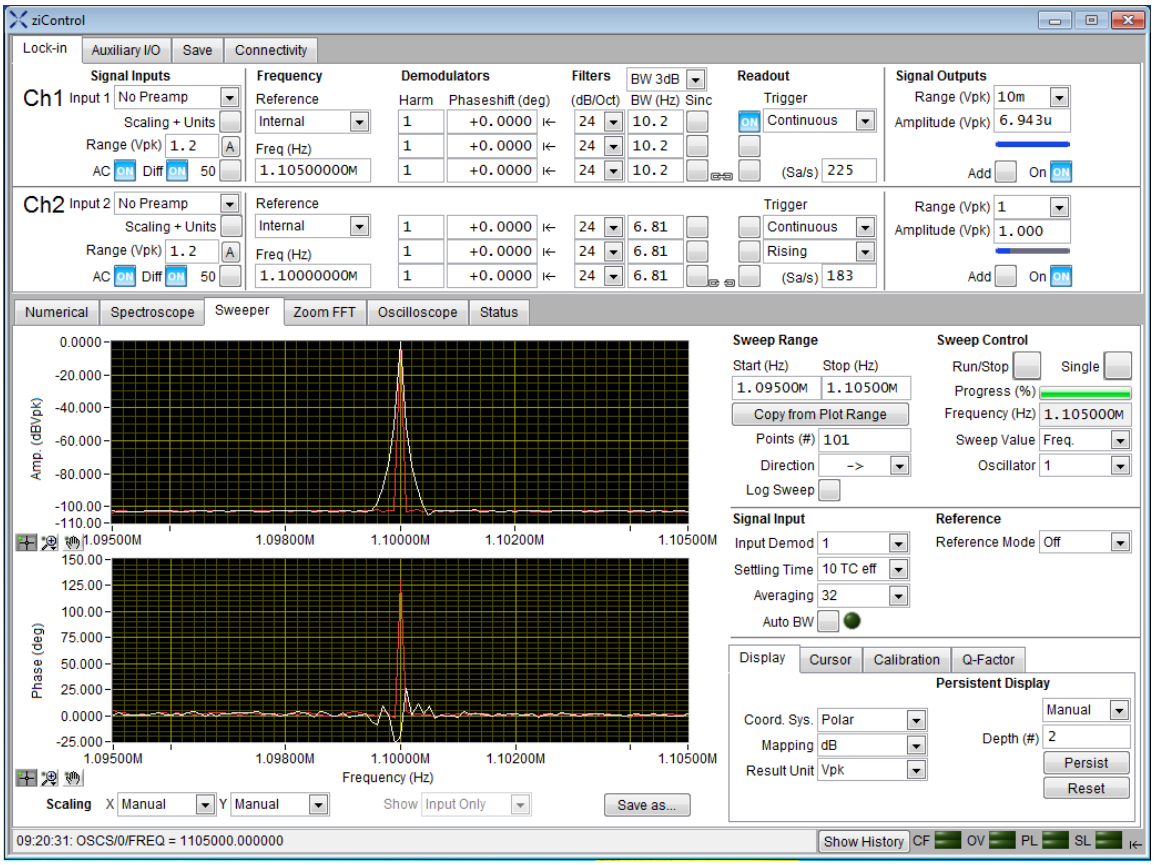


Figure 9.24. Dynamic reserve test results with Sweeper tool

When the disturbing frequency is equal to the measurement frequency, the DR is 0 dB, regardless of the filter settings. For larger filter bandwidths (white curve), the instrument is less capable of

rejecting the disturbing signal, than in the case of the narrower bandwidth (red curve). This is directly observed from the presented measurements. Note that the apparent width also depends on the sweep resolution. As the distance in frequency between the two signals increases, the DR increases as well. Sufficiently far away from the disturbing signal (this depends on the filter order and the BW), the DR can be calculated as: DR = 20*log($V_{disturb}$/ V_{meas}) = 20*log(1 V/ 6.943 µV) = 103.17 dB. To obtain the better separation; higher filter order, narrower bandwidth and additional averaging should be used. Note that the settling time increases with the filter order, which will result in longer measurement time.

In the second test, one may use the Spectroscope tool to verify that the accuracy of the measurement is as specified. Set the Ch1 and Ch2 reference frequency to 1.1 MHz and 1.09 MHz. Set the filter order to 48 dB/Oct and the time constant, TC eff to 2.3 s. Choose the Demod 1 in the Control section and set the Time scale to 50 s/Div. First measure the RMS of the 6.943 µV signal without the disturbance (Ch2 Signal Outputs amplitude should not be activated). Wait about 120 s and then apply the 1 V signal by pressing the ON button of the Ch2 Signal Outputs. Note the sharp increase of the signal amplitude. After the filter is settled (more than 15 TC) the measured signal converges to the expected RMS value of 4.91 µV. Stop the acquisition after about 120 s by pressing Acq Stop in the Spectroscope Control section. Activate cursors C1 and C2 to estimate the RMS amplitudes of the signal with and without the disturbance. Cursors C1 and C2 indicate estimated average RMS values before (4.889 µV) and after (4.936 µV) the disturbing signal has been applied. Both signals are measured with the 1% accuracy.

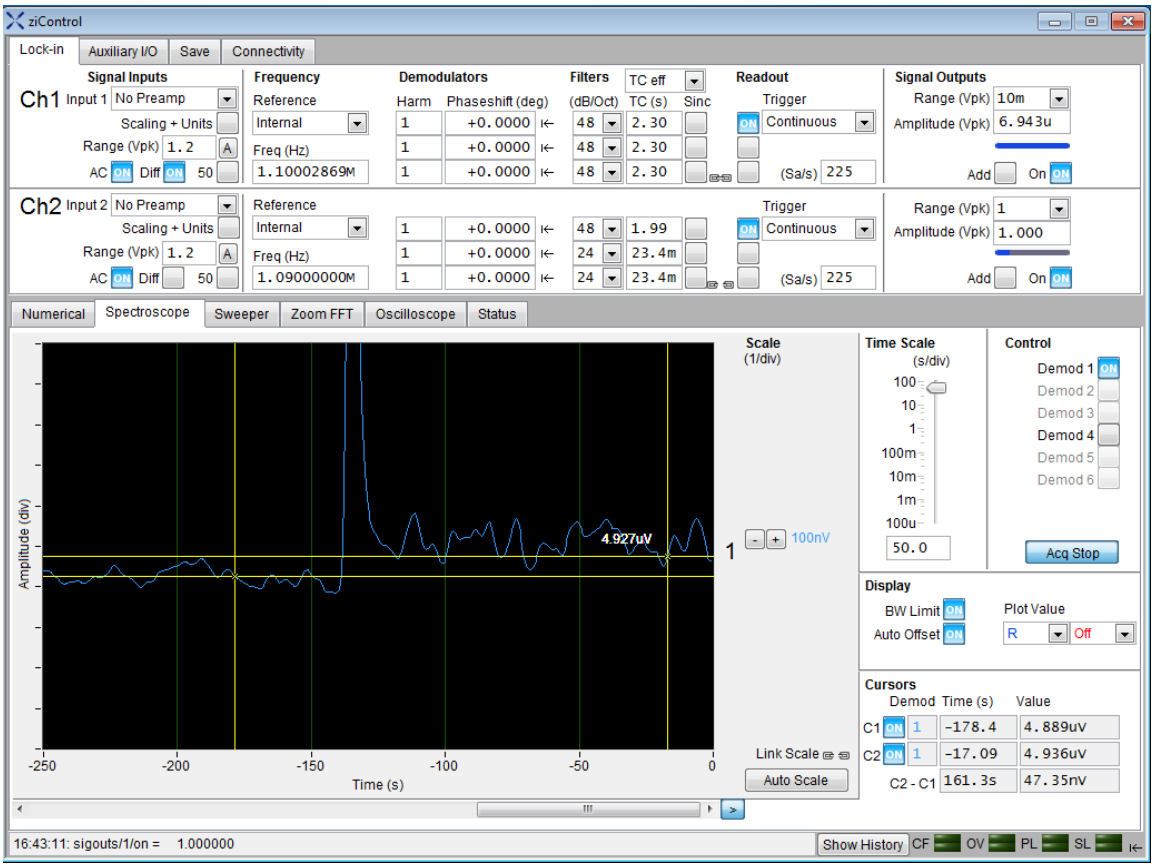


Figure 9.25. Dynamic reserve test results with Spectroscope tool

9.7.3. Test Common Mode Rejection Ratio

Definition

Common mode rejection ratio (CMRR) is the measure of the instrument's capability to reject signals that are common to both inputs (common mode). It is given in units of dB and expresses the ratio between the amplitude of the common mode interference (V_{cmm}) and the measured differential signal (V_{meas}),

$$CMRR = 20 * \log\left(\frac{V_{cmm}}{V_{meas}}\right) \tag{9.2}$$

Typical HF2 Instrument settings that influence the CMRR are the measurement frequency, the input range, and the selected input coupling (AC or DC).

Setup

Split the signal from the Output 1 Out and feed differentially to Input 1 (+ In and - Diff), using a T-connector and two short equal-length BNC cables. Make sure that the two BNC cables are not forming a large area loop, which is susceptible to the magnetic pick-ups. The setup is shown in the figure below.

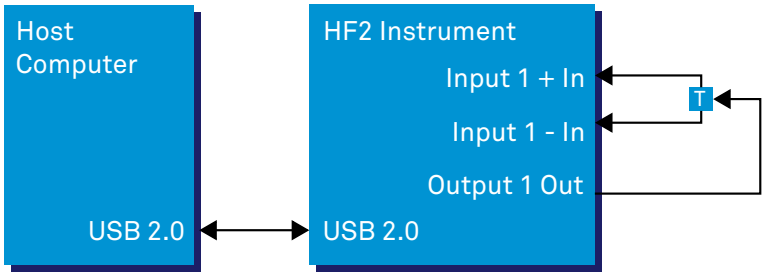


Figure 9.26. CMRR test setup

The HF2 instrument settings on the ziControl panel should be according to the table below:

Table 9.18. HF2 instrument settings

Ch1 Signal Input Range	100 mV
Ch1 Signal Input AC/ Diff/ 50	OFF/ ON/ OFF
Ch1 Demodulator 1 Readout	Cont.
Ch1 Signal Output Range	1.0 V
Ch1 Signal Output Amplitude	1.0 V
Ch1 Signal Output On	ON

Measurement

The measurements are performed using the frequency Sweeper with the following settings:

Table 9.19. Frequency Sweeper settings

Sweep Range Start	1 kHz
Sweep Range Stop	50 MHz
Sweep Range Points	100
Sweep Range Log Sweep	ON
Auto BW	OFF
BW 3dB	10 Hz
Settling Time	10 TC
Averaging	1
Display Mapping	dB
Result Unit	V_{RMS}
Display Polar	ON

After the settings have been applied, start the measurement by pressing the Sweep Control Single button. The plots of CMRR, as a function of frequency, in the range from 1 kHz to 50 MHz are displayed in the frequency sweeper. CMRR test examples with AC and DC coupling are shown below.

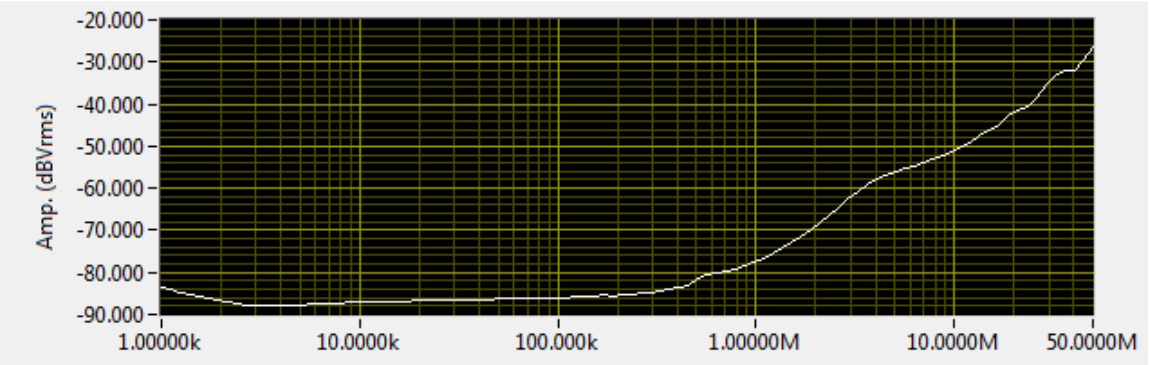


Figure 9.27. CMRR test results with AC coupling

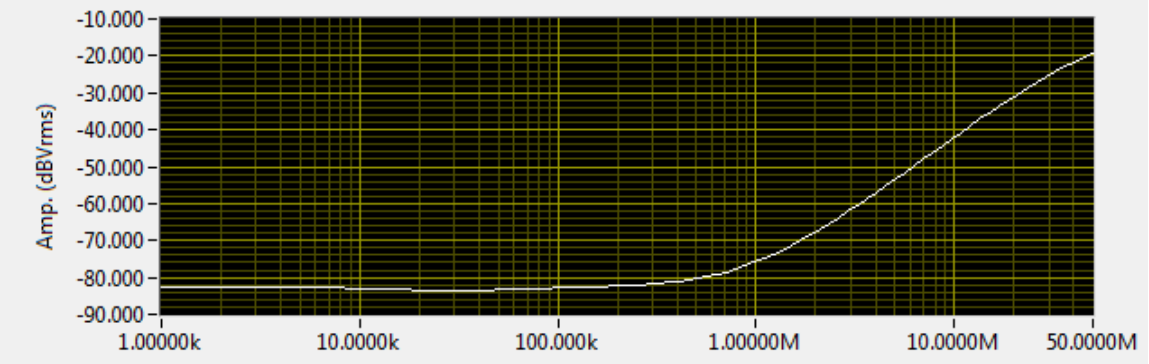


Figure 9.28. CMRR test results with DC coupling

The test shows that the common mode is rejected at, and below 75 dB, in the frequency range from 1 kHz to 1MHz and gradually decreases as the frequency increases to 50 MHz.

9.7.4. Test Total Harmonic Distortion

Definition

Total harmonic distortion (THD) is a measure of the non-linearity of the input and output channels. It is given in units of dB and expresses the ratio of the root-mean-square, of all higher harmonics ($H_{n>1}$), and the amplitude of the fundamental frequency (H_1),

$$\mathrm{THD} = 20*\log\left(\frac{\sqrt{\sum_{n=2}^{\infty} H_n^2}}{H_1}\right) \tag{9.3}$$

The HF2 Instruments allows the user to measure harmonics up to n = 1023, limited by the bandwidth of the instrument. However, to determine the THD, it is sufficient to measure the first few significant harmonics (up to n = 5) and neglect the higher order ones. With the HF2 Instrument, simultaneous measurements of up to three harmonics can be done per each channel. The THD can be estimated using the simplified formula:

$$\mathrm{THD} = 20*\log\left(\frac{\sqrt{\sum_{n=2}^{5} H_n^2}}{H_1}\right) \tag{9.4}$$

Typical HF2 Instrument settings that influence the THD are the input and output ranges, the measurement frequency, and the selected input coupling (AC or DC).

Setup

Connect the Signal Output1 Out to the Signal Input 1 +In using a short BNC cable. The setup is shown in the figure below.

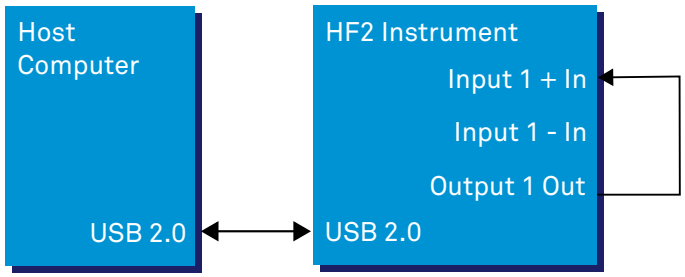


Figure 9.29. THD test setup

The HF2 instrument settings on the ziControl panel should be according to the table below:

Table 9.20. HF2 instrument settings

Ch1 Signal Inputs Range	1.0 V
Ch1 Signal Inputs AC/ Diff / 50	OFF/ OFF /OFF
Ch1 Frequency (Hz)	1 MHz

Ch1 Demodulators 1/ 2 /3 Harm	1/ 2/ 3 (1/ 4/ 5)
Ch1 Demodulators 1/ 2 /3/ Osc	1
Ch1 Filters 1/ 2 /3 Input	1
Ch1 Filters 1/ 2 /3 BW	1 Hz
Ch1 Filters 1/ 2 /3 dB/Oct	24
Ch1 Demodulators 1/ 2 /3 Readout	Cont.
Ch1 Signal Output Range	1 V
Ch1 Signal Output Amplitude	100 mV

Measurement

The user should perform the test in two or more steps to make sure that all the significant harmonics are included. Measure the first three harmonics 1, 2, and 3 in the first step. Continue with harmonics 1, 4, and 5 in the second step. Calculate the THD using the simplified formula given in the definition section.

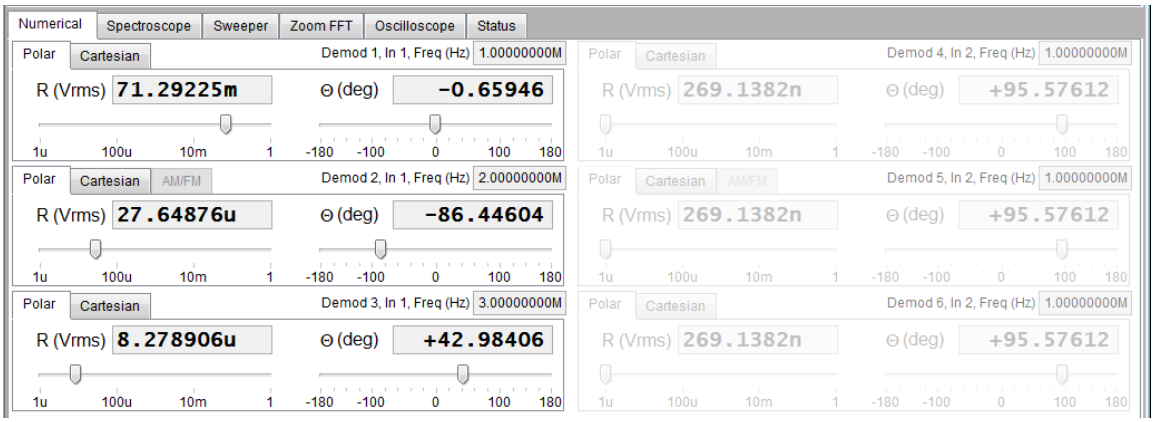


Figure 9.30. THD test results for harmonics n=1, 2, 3

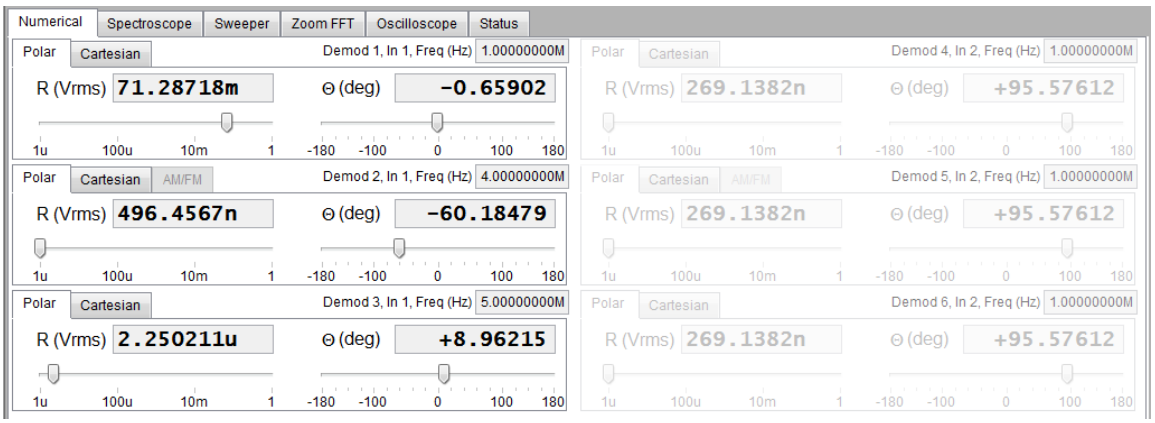


Figure 9.31. THD test results for harmonics n=1, 4, 5

Using the first five harmonic amplitudes, measured at 1 MHz, the THD of the HF2 instruments input and output is approximately - 68 dB.

9.7.5. Test Lock-time

Definition

The lock-time of a lock-in amplifier is the time the instrument takes to react on a sudden change of the external reference frequency. This is technically equivalent to the time the internal phase-locked loop takes to reach the locked state starting from an unlocked state. It is usually given as the number of reference signal cycles (periods) or in seconds. The lock-time can be calculated as, $T_{lock\text{-}time}=T_{lock}-T_{unlock}$, where T_{unlock} is the time at which an external signal with frequency f_s is applied at the signal inputs, and T_{lock} is the time after which the reference frequency is within the 0.5% of f_s.

The HF2 Instrument graphical panel also displays a locked state indicator which turns green when additional phase error conditions are met.

Typical HF2 Instrument settings that influences the lock-time depends on the external reference frequency and amplitude, the input range.

Setup

Connect the Signal Output 2 Out connector to the Signal Input 1 +In connector using a short BNC cable. The setup is shown in the figure below.

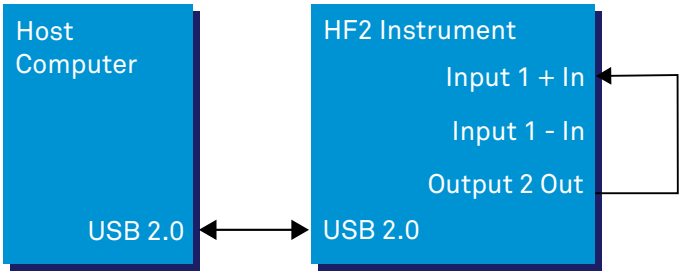


Figure 9.32. Lock-time test setup

The HF2 Instrument settings on the ziControl panel should be according to the table below:

Table 9.21. HF2 instrument settings

Ch1 Signal Inputs Range	200 mV
Ch1 Signal Inputs AC/ Diff/ 50	ON/ OFF/ OFF
Ch1 Reference	Signal In 1 (auto)
Ch1 Frequency	45 Hz
Ch1 Filters dB/Oct	24
Ch1 Filters BW	1 Hz
Ch1 Demodulator 1 Readout	28.1
Ch1 Demodulators 1	ON
Ch2 Outputs range	1 V
Ch2 Outputs	ON

Ch2 Outputs Amplitude	0 V, (unlocked state)
Ch2 Outputs Amplitude	100 mV (at T_{unlock}, external signal applied)
Ch2 Frequency (Hz)	50 Hz

Measurement

The measurements are performed using the Spectroscope tool with the following settings:

Table 9.22. HF2 Spectroscope settings

Time Scale	1.00 s/Div
Control	Demod 1
Display Plot Value	R, Freq
Cursors	C1, C2

Observe the Demod 1 values (R and Freq.) vs time for a few seconds and then change the amplitude of Ch2 from 0 V to 0.1 V. Stop the acquisition when the frequency is settled to the 50 Hz value and after the locked indicator turned green. Use the activated cursors to mark the onset of the external signal application as well as the position in time where the frequency has settled. In this particular case the estimated lock time is $T_{lock\text{-}time}$= 1.21 s, as shown in the figure below, which is approximately 60 cycles.

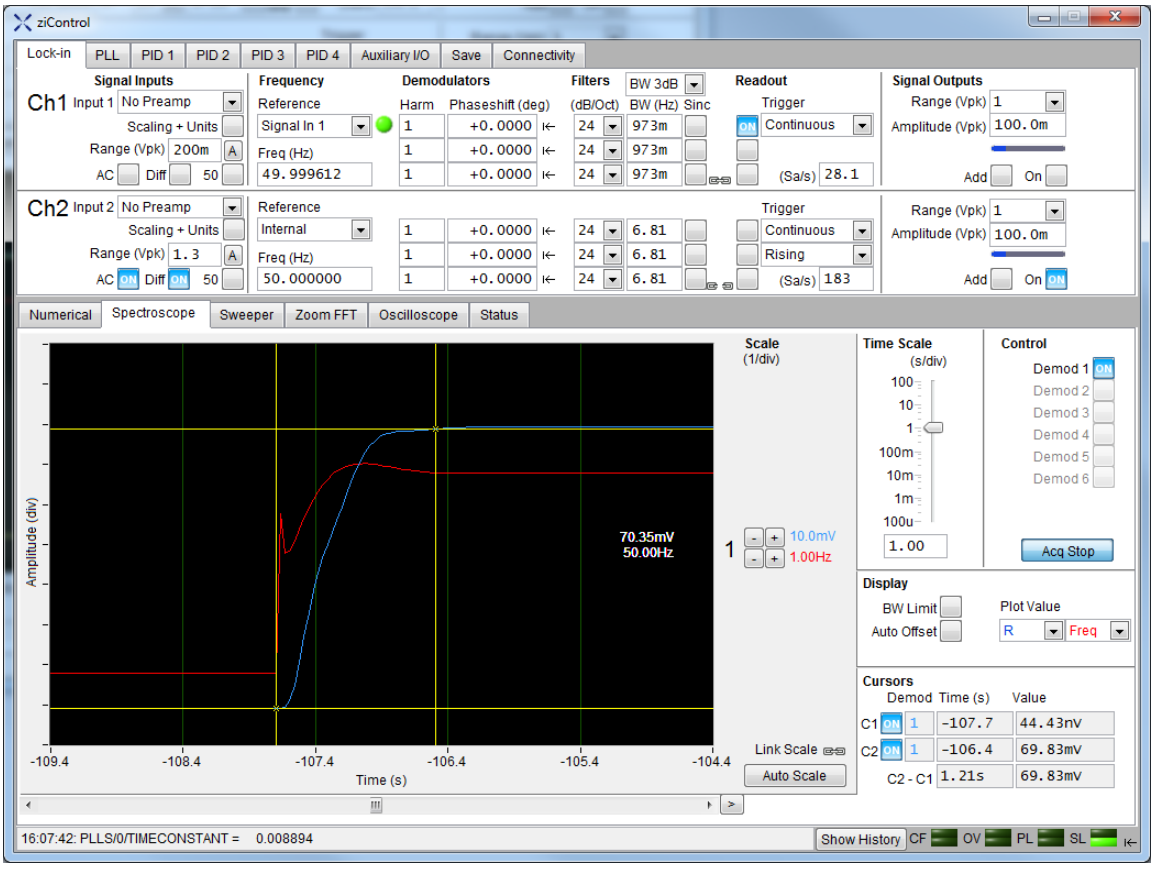


Figure 9.33. Lock-time test results for frequency of 50 Hz

Note that the Spectroscope tool is suitable for monitoring the HF2 instrument's locking to low frequencies, where the lock-time is larger. For external frequencies higher than 1 MHz, user is advised to save the desired demodulator data and plot demodulator frequency as a function of time. Make sure that the sampling rate is increased accordingly with the signal frequency to obtain sufficient time stamp resolution. The PLL lock-time can be then determined similarly to the procedure described above.

9.7.6. Test Frequency Accuracy

Definition

Frequency accuracy (FA) is a measure of an instrument's ability to faithfully indicate (f_m) the correct frequency versus a traceable standard of an internal or external reference signal (f_r). It is usually given in ppm (parts per million) and can be calculated using the expression,

$$\mathrm{FA(ppm)} = \frac{f_r - f_m}{f_m} * 1'000'000 = \frac{f_e}{f_r} * 1'000'000 \quad (9.5)$$

where, $f_e = f_r - f_m$ is the frequency measurement error.

Setup

Connect the output of a function generator to the Input 2 + In, using a short BNC cable. The function generator should have at least 3 ppm frequency accuracy. The setup is shown in the figure below.

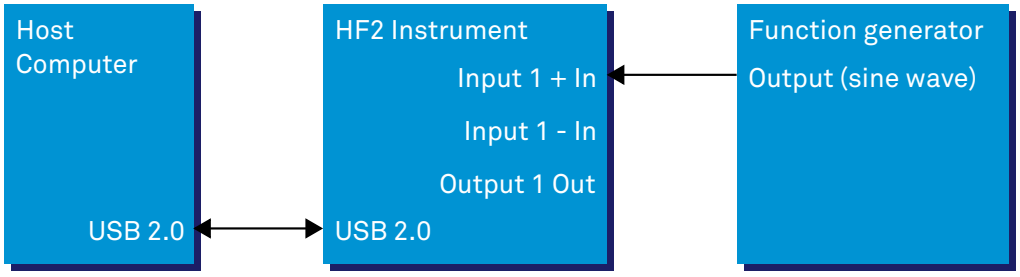


Figure 9.34. Frequency accuracy test setup

The HF2 instrument settings on the ziControl panel should be set according to the table below:

Table 9.23. HF2 instrument settings

Ch2 Signal Inputs Range	1.0 V
Ch2 Signal Inputs AC/ Diff/ 50	OFF/ OFF/ OFF
Ch2 Frequency Reference	Signal Input 2

Measurement

The measurement procedure consists of the following steps:

1. Set the amplitude of the function generator to 500 mV.
2. Set the reference frequency, f_r, of the function generator using the table below, or any other frequency of interest.
3. Wait for the first five digits of the measured frequency,f_m, to stabilize.
4. Read out the displayed reference frequency and calculate the error according to $f_e=f_r-f_m$.

   Example: For the reference frequency f_r = 10 kHz measured frequency is f_m = 9.99991 kHz. The frequency error is thus, f_e= 0.09 Hz. Finally, frequency accuracy FA = f_e /f_m*1'000'000 = 0.09 Hz/10 kHz *1'000'000 = 9 ppm ~ 10 ppm.

   Repeat steps 2-4 using frequencies given in the table below.

Table 9.24. Frequency accuracy test table

Reference Frequency	Measured Frequency	Frequency accuracy
1 kHz		
10 kHz		
100 kHz		
1 MHz		
10 MHz		

9.7.7. Test Output Amplitude Accuracy

Definition

The output amplitude accuracy (OAA) accuracy is a measure of an instrument's ability to faithfully output a set voltage at a given frequency. It is usually given as the percentage, %, of the measurement deviation from a traceable standard, and can be calculated using the expression,

$$\mathrm{OAA}\,(\%)=\frac{\mathrm{OA_{expect}}-\mathrm{OA_{meas}}}{\mathrm{OA_{expect}}}*100 \tag{9.6}$$

Typical HF2 Instrument settings which influence output amplitude accuracy are the output range and the frequency of the output signal. Above a certain frequency the OAA is heavily impacted by the output filters.

Setup

Connect the Output 1 Out of the HF2 to the digital multimeter using a short BNC cable. The total digital multimeter measurement error should be at least 10 times smaller than HF2's measurement error in order for it to be suitable as the test device. The digital multimeter should be adjusted to measure AC voltage. The setup is shown in the figure below.

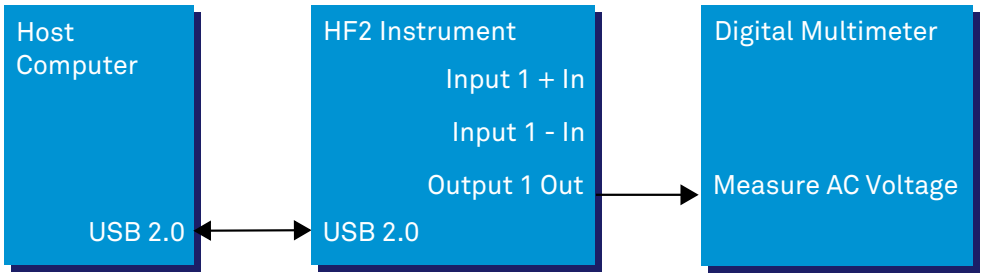


Figure 9.35. Output amplitude accuracy test setup

The HF2 Instrument settings on the ziControl panel should be according to the table below:

Table 9.25. HF2 instrument settings

Ch1 Signal Output Range	Variable
Ch1 Signal Output Amplitude	Variable
Ch1 Signal Output On	ON
Ch1 Frequency (Hz)	10 kHz (or other, subject to multimeter bandwidth)

Measurement

The measurement procedure consists of the following steps:

1. Select the HF2 channel output range.
2. Set the output amplitude, OA_{expect}, to the 10 %, followed by 50 % and 80 % of the output range selected in step 1.
3. Record multimeter reading, OA_{meas}.
4. Calculate OAA using the formula given in the definition section.
5. Calculate total measurement accuracy of the digital multimeter. The total multimeter reading error is specified in the multimeter user manual and an example of how to calculate it is shown below.

   Example: the reading error of the set amplitude 80 mV and measured 79.95 mV is 0.0006 * 79.95 mV, plus the range error of 0.0003*100 mV. This gives the total measurement error of ± 0.07797 mV and a total multimeter measurement accuracy of 0.07797 mV/ 80 mV *100 = 0.09746 % ~ 0.1 %.
6. Compare the digital multimeter measurement accuracy to that of OA. It should be at least an order of magnitude better than the OAA of the HF2 Instrument. If this is not the case, this multimeter cannot be used for accuracy measurements in this particular range.
7. Go through steps 1-6 for all the output ranges, as indicated in the test table.

Table 9.26. Output amplitude accuracy test table

Output range	Output amplitude	Multimeter reading	Accuracy (%)
1 V			

100 mV			
10 mV			

9.7.8. Test Input Amplitude Accuracy

Definition

The input amplitude accuracy (IAA) is a measure of instrument's capability to faithfully indicate the signal amplitude at the input channel. It is given usually as percentage, %, of the measurement deviation from a traceable standard, and can be calculated using the expression,

$$\mathrm{IAA}\,(\%) = \frac{\mathrm{IA}_{\mathrm{expect}} - \mathrm{IA}_{\mathrm{meas}}}{\mathrm{IA}_{\mathrm{expect}}} * 100 \tag{9.7}$$

Typical HF2 Instrument settings that influence input amplitude accuracy are the input range, measurement frequency, input impedance and selected coupling (AC or DC). Above a certain frequency, the IAA is heavily impacted by the input anti-aliasing filters.

This parameter is also called input gain accuracy.

Setup

Use a T-connector to split the output of a function generator and feed it into the HF2 Signal Input 1 +In, and a digital multimeter. Adjust the digital multimeter for AC voltage measurements. The digital multimeter's total measurement error should be at least 10 times smaller than HF2's measurement error in order for it to be suitable as the test device. In addition, multimeter's bandwidth for AC voltage measurements is usually limited to 10 or 20 kHz. To measure the accuracy at higher frequencies, a spectrum analyzer may be used. The setup is shown in the figure below.

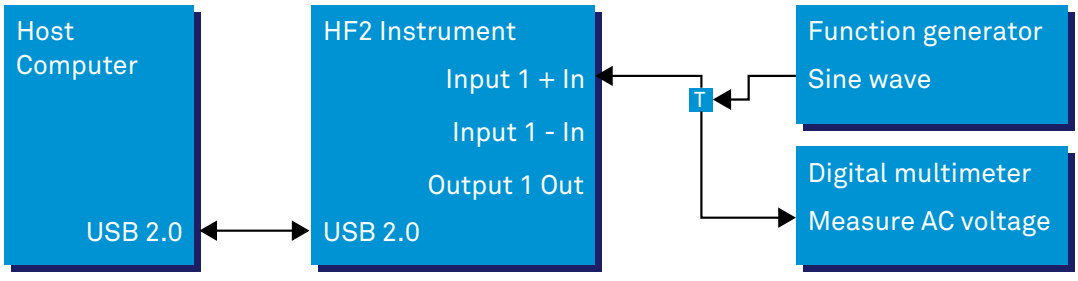


Figure 9.36. Input amplitude accuracy test setup

The HF2 instrument settings on the ziControl panel should be set according to the table below:

Table 9.27. HF2 instrument settings

Ch1 Signal Inputs Range	Variable
Ch1 Signal Inputs AC/ Diff/ 50	OFF/ OFF/ OFF
Ch1 Frequency Reference	Signal Input 1
Ch1 Demodulators 1 Harm	1
Ch1 Demodulators 1 Readout	Cont.
Ch1 Filters TC (s)	100 ms
Ch1 Filters dB/Oct	48

Measurement

The measurement procedure consists of the following steps:

1. Select the HF2 input channel range.
2. Set the frequency of the function generator to 10 kHz and amplitudes to the 10 %, followed by 50% and 80% of the input range set in step 1.
3. Wait for the measured signal to settle and read out the RMS amplitudes of the HF2 Demodulator 1, IA_{meas}, and the digital multimeter, IA_{expect}.
4. Calculate IA accuracy according to the formula given in the definition section.
5. Calculate total measurement error with given accuracy of the digital multimeter. The total multimeter reading error is specified in the multimeter user manual and an example of how to calculate it is shown below.

   Example: for 100 mV amplitude whose RMS amplitude is 70.71 mV, digital multimeter reading error is 0.0006 * 71.07 mV, plus the range error of 0.0003 * 100 mV. This gives the total measurement error of ± 0.0726 mV and accuracy of 0.0726 mV/100 mV*100 ~ 0.1 %.
6. Compare the digital multimeter measurement accuracy to that of IA. It should be at least an order of magnitude better than that of HF2 IA accuracy. If this is not the case, this multimeter cannot be used for accuracy measurements in this particular range.
7. Repeat steps 1-6 for all input ranges, as indicated in the test table.

Table 9.28. Input amplitude accuracy test table

Input range	Multimeter reading	HF2 reading	Accuracy (%)
1 V			
100 mV			
10 mV			

Chapter 10. Signal Processing Basics

This chapter provides insights about several lock-in amplifier principles with a generalized perspective not necessarily linked to a specific instrument from Zurich Instruments. Since the appearance of the first valve-based lock-in amplifiers in the 1930s the physics have not changed, but the implementation and the performance have seen several step functions. Many good lock-in amplifier primers have appeared in the past decades, and some of them appear outdated now because they were written with analog instruments in mind. This section does not aim to replace any existing primer, but to complete them with a preferred emphasis on digital lock-in amplifiers.

The first subsection describes the basic principles of lock-in amplification, followed by the description of the function of time discrete filters. After, a specification particularly important for analog lock-in amplifier is described, and the reason why the significance for digital lock-in amplifier is reduced is explained. Sinc filtering is subject that has seldom been described in connection with lock-in amplifiers also deserves its own sections.

Zoom FFT is a feature that Zurich Instruments considers innovative in the context of lock-in amplifiers and therefore deserves a mathematical justification and explanation.

10.1. Principles of Lock-in Detection

Lock-in demodulation is a technique that permits to measure the amplitude A_S and the phase Θ_S of a periodic signal with the frequency $\omega_S = 2 * \pi * f_S$ by comparing it to a reference signal. This technique is also referred to as phase-sensitive detection. By averaging over time the signal-to-noise ratio (SNR) of a signal can be increased by orders of magnitude, allowing very small signals to be detected with a high accuracy making the lock-in amplifier a tool often used for signal recovery. For both signal recovery and phase-sensitive detection, the signal of interest is isolated with narrow band-pass filtering therefore reducing the impact of noise in the measured signal.

Figure 10.1 shows a basic measurement setup: a reference V_R signal is fed to the device under test. This reference signal is modified by the generally non-linear device with attenuation, amplification, phase shifting, and distortion, resulting in a signal $V_S = A_S \cos(\omega_S t + \Theta_S)$ plus harmonic components.

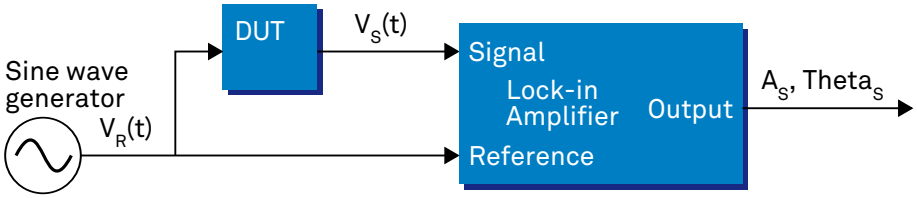


Figure 10.1. Basic measurement setup incorporating a lock-in amplifier

For practical reasons, most lock-in amplifiers implement the band-pass filter with a mixer and a low-pass filter (depicted in Figure 10.2): the mixer shifts the signal of interest into the baseband, ideally to DC, and the low-pass filter cuts all unwanted higher frequencies.

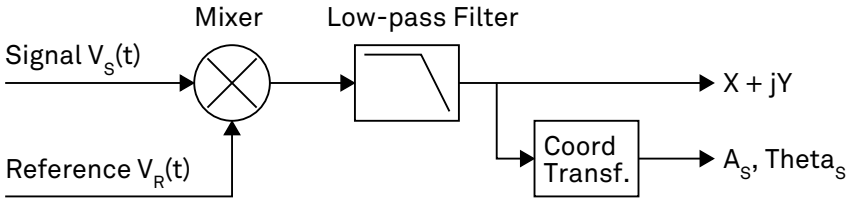


Figure 10.2. Mixing and low-pass filtering performed by the lock-in amplifier

The input signal $V_S(t)$ is multiplied by the reference signal $V_R(t) = \sqrt{2}\, e^{-j\omega_R t}$, where $\omega_R = 2 * \pi * f_R$ is the demodulation frequency. This is the complex representation of a sine and cosine signal (phase shift 90°) forming the components of a quadrature demodulator, capable to measure both the amplitude and the phase of the signal of interest. In principle it is possible to multiply the signal of interest with any frequency, resulting in a heterodyne operation. However the objective of the lock-in amplifier is to shift the signal as close as possible to DC, therefore the frequency of the reference and the signal is chosen similar. In literature this is called homodyne detection, synchrodyne detection, or zero-IF direct conversion.

The result of the multiplication is the signal

$$V_S(t) * V_R(t) = V_S(t) * \sqrt{2}\, e^{-j\omega_R t} = \frac{A_S}{\sqrt{2}} e^{+j[(\omega_S-\omega_R)t+\Theta_S]} + \frac{A_S}{\sqrt{2}} e^{-j[(\omega_S+\omega_R)t+\Theta_S]}$$

Equation 10.1. Multiplication of signal of interest with reference signal

which contains a slow signal component with frequency f_S-f_R and a fast signal component with frequency f_S+f_R.

The demodulated signal is then low-pass filtered with an infinite response RC filter, indicated by the symbol $\langle \cdot \rangle$. The frequency response of the filter F(ω) will let pass the low frequencies F(ω_S-ω_R) while considerably attenuating the higher frequencies F(ω_S+ω_R). Another way to consider the low-pass filter is an averager, that performs cuts the higher components in the signal.

$$X + \mathrm{jY} = \left\langle V_S(t) * \sqrt{2}\, e^{-j\omega_R t} \right\rangle \approx F(\omega_S - \omega_R) \frac{A_S}{\sqrt{2}} e^{j[(\omega_S - \omega_R)t + \varphi_S]}$$

Equation 10.2. Averaging the result of the signal multiplication

The result after the low-pass filter is the demodulated signal $X + \mathrm{jY}$, where X is the real and Y is the imaginary part of a signal depicted on the complex plane. These components are also called in-phase and quadrature components. The transformation of X and Y into the amplitude R and phase Θ information of V_S(t) can be performed with trigonometric operations.

It is interesting to note that the value of the measured signal corresponds to the RMS value of the signal, which is equivalent to R = A_S / √2.

Most lock-in amplifiers output the values (X,Y) and (R, Θ) encoded in a range of -10 V to 10 V of the auxiliary output signals.

10.1.1. Lock-in Amplifier Applications

Lock-in amplifiers are employed in a large variety of applications. In some cases the objective is measuring a signal with good signal-to-noise ratio, and then that signal could be measured even with large filter settings. In this context the word phase sensitive detection is appropriate. In other applications, the signal is very weak and overwhelmed by noise, which forces to measure with very narrow filters. In this context the lock-in amplifier is employed for signal recovery. Also, in another context, a signal modulated on a very high frequency (GHz or THz) that cannot be measured with standard approaches, is mixed to a lower frequency that fits into the capabilities of the lock-in amplifier.

One example for measuring a small pseudo-DC signal (meaning, that it does not vary in time very much) which is completely buried in the 1/f noise and 50/60 Hz induction. For this purpose a weak signal is modulated to a higher frequency less subject to noise. Such signal can be efficiently mixed back and measured in the baseband using a lock-in amplifier. In Figure 10.3 this process is depicted. Many optical applications perform the up-mixing with a chopper, an electro-optical modulator, or an acousto-optical modulator. The advantage of this procedure is that the desired signal is measured in a spectral region that is relatively free of noise. This is more efficient than just low pass filtering the DC signal.

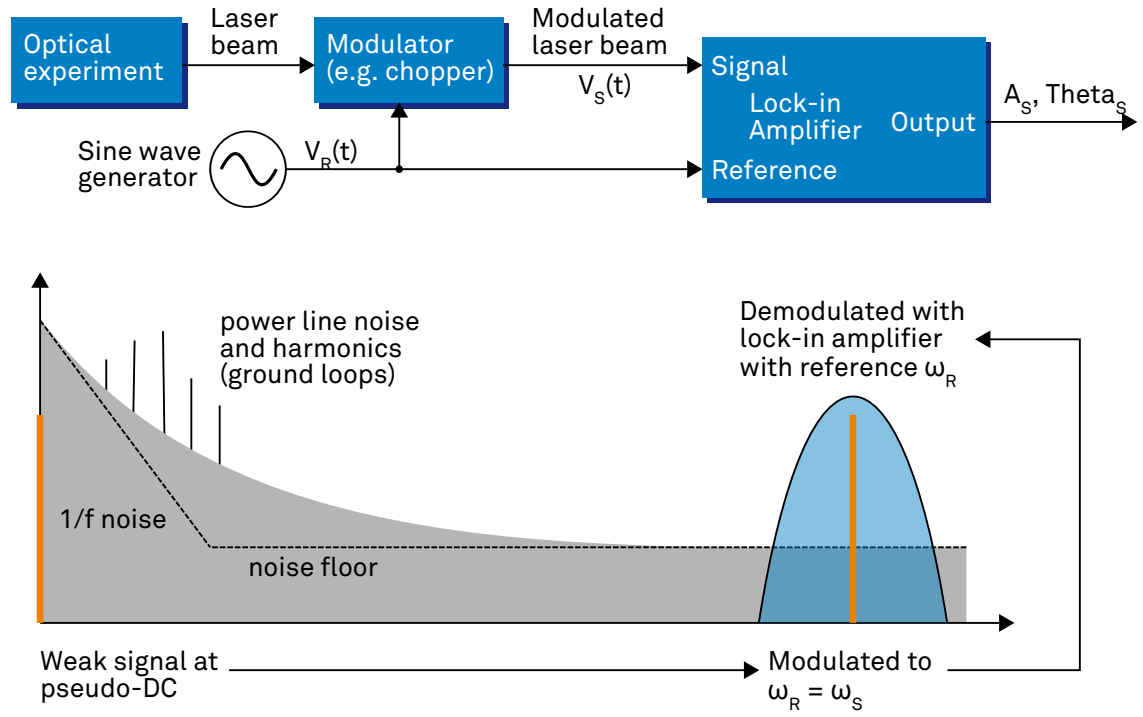


Figure 10.3. DC signal with low noise

10.2. Signal Bandwidth

The signal bandwidth (BW) theoretically corresponds to the highest frequency components of interest in a signal. In practical signals, associated with the signal BW, the cut-off frequency is defined where the transfer function of a system shows 3 dB attenuation versus DC (BW = $f_{cut\text{-}off}$ = f_{-3dB}). The bandwidth, equivalent to cut-off frequency, is used in the context of dynamic behavior of a signals or separation of different signals. This is for instance the case for fast changing amplitudes or phase values like in a PLL or in a imaging application, or when signals close in frequency need to be separated.

The noise equivalent power bandwidth (NEPBW) is also a useful figure, and it is distinct from the signal bandwidth. This unit is typically used for noise measurements: in this case one is interested in the total amount of power that goes through a low-pass filter, equivalent to the blue area in the figure below. For practical reasons, the blue area is converted into an ideal brickwall filter, a filter with transmission 1 from DC to f_{NEPBW}, depicted as the orange area. The low-pass filter and the brickwall filter of Figure 10.4 let pass the same amount of power under the assumption that the noise has a uniform (white) distribution.

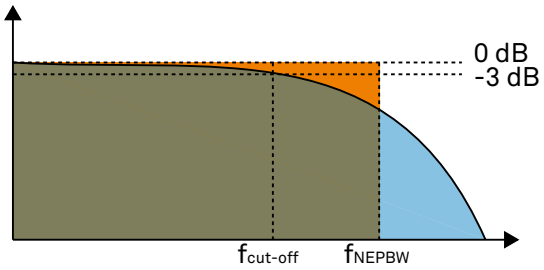


Figure 10.4. Signal BW versus NEPBW

It is possible to establish a simple relation between the $f_{cut\text{-}off}$ and the f_{NEPBW} that only depends on the slope (or roll-off) of the filter. As the filter slope actually depends on the time constant (TC) defined for the filter, it is possible to establish the relation also to the TC. It is intuitive to understand that for higher filter orders, the $f_{cut\text{-}off}$ is closer to the f_{NEPBW} than for smaller orders.

The time constant of a low pass filter relates to the bandwidth according to the formula

$$\mathrm{BW} = \frac{\mathrm{FO}}{2*\pi*\mathrm{TC}} \tag{10.3}$$

where FO is said factor that depends on the filter slope. The time constant is a parameter used to interpret the filter response in the time domain, and relates to the time it takes to reach a defined percentage of the final value.

Table 10.1. Summary of conversion factors for bandwidth definitions

filter order	filter roll-off	FO	$f_{cut\text{-}off}$	f_{NEPBW}	f_{NEPBW} / $f_{cut\text{-}off}$
1st	6 dB/oct	1.000	0.159 / TC	0.250 / TC	1.57
2nd	12 dB/oct	0.644	0.102 / TC	0.125 / TC	1.22
3rd	18 dB/oct	0.510	0.081 / TC	0.094 / TC	1.15
4th	24 dB/oct	0.435	0.068 / TC	0.078 / TC	1.12
5th	30 dB/oct	0.386	0.062 / TC	0.068 / TC	1.11
6th	36 dB/oct	0.350	0.056 / TC	0.062 / TC	1.10
7th	42 dB/oct	0.323	0.051 / TC	0.056 / TC	1.10

filter order	filter roll-off	FO	$f_{cut\text{-}off}$	f_{NEPBW}	f_{NEPBW} / $f_{cut\text{-}off}$
8^{th}	48 dB/oct	0.301	0.048 / TC	0.052 / TC	1.09

10.3. Discrete Filter

There are many options how to implement digital low pass filters. One common way is the cascaded RC filters: for instance the 4th order filter is implemented by chaining 4 filters with the same time constant TC = τ_N one after the other so that the input of the next stage of filtering is the output of the previous one. The time domain algorithm for a single stage is described as

$$X_{\text{out}}[n,T_S] = e^{-T_S/\tau_N} X_{\text{out}}[n-1,T_S] + (1-e^{-T_S/\tau_N}) X_{\text{in}}[n,T_S]$$

Equation 10.4. Time domain response of the discrete-time RC filter

where $X_{\text{in}}[n,T_S]$ is the nth input signal, $X_{\text{out}}[n,T_S]$ the nth output signal and T_S the sampling time. In the frequency domain, a good approximation of its transfer function is

$$H(\omega) = \frac{1}{1+j*\omega*\tau_N}$$

Equation 10.5. Frequency domain response of the discrete-time RC filter

The attenuation and phase shift of the filters can be obtained from this formula.

10.3.1. Filter Settling Time

The low pass filters after the demodulator cause a delay to measured signals depending on the filter order and time constant TC = τ_N. Therefore, it will take the lock-in some time before the correct final value is reached. This is depicted in Figure 10.5 where the time response of cascaded filters to a step input signal this is shown.

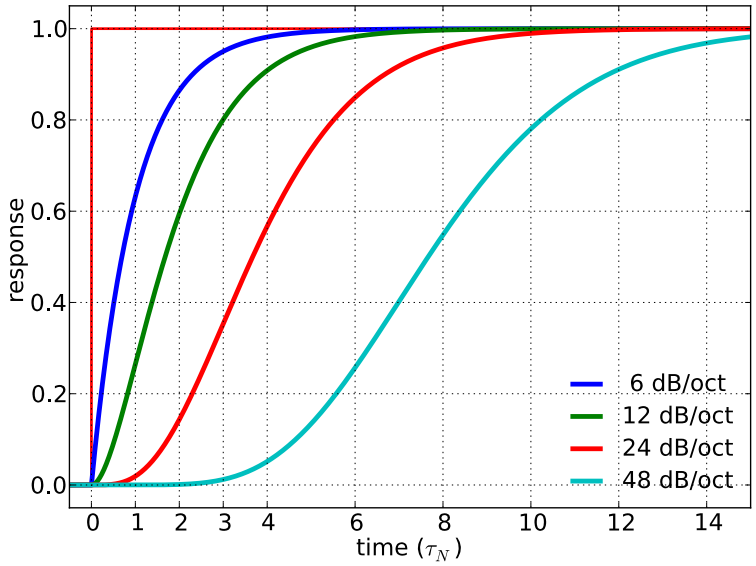


Figure 10.5. Time domain step response of the RC low pass filters

The same information can also be displayed in Table 10.2 as a function of time constants to read a defined percentage of the final value. For the filter orders obtainable with the HF2 series devices, you find delay values in units of the filter time constant τ_N in. The values tell, the time you need to wait for the filtered demodulator signal to reach 5%, 95% and 99% of the correct value.

Table 10.2. Summary of Filter Rise Times

filter order	Setting time to		
	5%	95%	99%
1st	0.025 * TC	3.0 * TC	4.6 * TC
2nd	0.36 * TC	4.7 * TC	6.6 * TC
3rd	0.82 * TC	6.3 * TC	8.4 * TC
4th	1.4 * TC	7.8 * TC	10 * TC
5th	2.0 * TC	9.2 * TC	12 * TC
6th	2.6 * TC	11 * TC	12 * TC
7th	3.3 * TC	12 * TC	15 * TC
8th	4.0 * TC	13 * TC	16 * TC

10.4. Full Range Sensitivity

The sensitivity of the lock-in amplifier is the RMS value of an input sine that is demodulated and results in a full scale analog output. Traditionally the X, Y, or R components are mapped onto the 10 V full scale analog output. In such a case, the overall gain from input to output of the lock-in amplifier is composed by the input and output amplifier stages. Many lock-in amplifiers specify a sensitivity between 1 nV and 1V. In other words the instrument permits an input signal between 1 nV and 1 V to be amplified to the 10 V full range output.

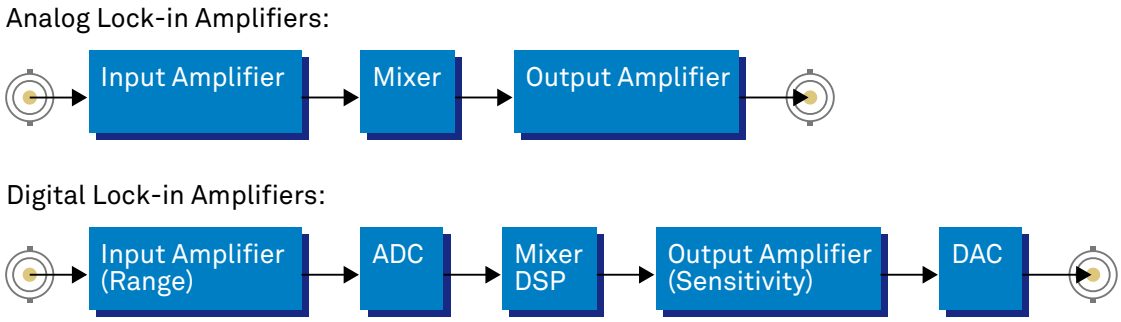


Figure 10.6. Sensitivity from signal input to signal output

In analog lock-in amplifiers the sensitivity is simple to understand. It is the sum of the analog amplification stages between in the input and the output of the instrument: in particular the input amplifier and the output amplifier.

In digital lock-in amplifiers the sensitivity is more complex. Analog to digital converters operate with a fixed input range (e.g. 1V) and thus require a variable gain amplifier to amplify the input signal to the range given by the ADC. This variable gain amplifier must be in the analog domain and its capability determines the minimum input range of the instrument. A practical analog input amplifier provides a factor 1000 amplification, thus 1 V divided by 1000 is the minimum input range of the instrument.

The input range is not the sensitivity. The input range is the maximum signal amplitude that is permitted for a given range setting. The signal is internally amplified with the suited factor (1 mV -> 1000) to result in a full swing signal at the ADC. For signals larger than the range, the ADC saturates implying a distortion of the signal and destroying the measurement result. Thus the signal should never exceed the range setting.

The sensitivity is actually not directly dependent from the input range. In digital lock-in amplifiers the sensitivity is only determined by the output amplifier, which performs a numerical multiplication of the demodulator output with the scaling factor. It is this scaling factor that can be retrofitted to specify a sensitivity as known from the analog lock-in amplifiers.

One interesting aspect of signal digitalization should be considered hereafter. As the ADC operates with a finite resolution, for instance 14 bits, the minimum signal that can be detected and digitized is for instance 1 mV divided by the resolution of the ADC. With 14 bits the minimum level that can be digitized would be 122 nV. How is it possible to reach 1 nV sensitivity without using a 21 bit analog to digital converter (not possible with 2010 technology at speed above 100 MHz)? In a world without noise it is not possible. Inversely, thanks to noise and current digital technology it is possible to achieve a sensitivity even below 1 nV.

The solution is noise. Gaussian noise is equally distributed in a signal, and thus generates equally distributed disturbances. The RMS average of the noise is however a DC value and which can be filtered by the lock-in amplifier and does not impact the measurement. However thanks to the presence of random noise (100 nV and more), the ADC is able to toggle the least significant bits

even for very small signals, as low as 1 nV and below. The occasional toggling has a low frequency that can be detected by the lock-in amplifier.

An example from biology. Rod cells in the human eye permit humans to see in very low light conditions. The sensitivity of rod cells in the human eye is as low as a single photon. This sensitivity is achieved in low light conditions by a sort of pre-charging of the cell to be sensitive to the single photon that triggers the cell to fire an impulse. In a condition with more surround light, rod cells are less sensitive and need more photons to fire.

Summarizing: in digital lock-in amplifiers the full range sensitivity is only determined by the scaling factor capability of the digital output amplifier. As the scaling can be arbitrary big, 1 nV minimum full range sensitivity is achievable without a problem. Further, digital lock-in amplifiers exploit the high frequency components of noise to heavily increase the sensitivity without impacting the accuracy of the measurement.

10.5. Sinc Filtering

It is a fact of lock-in amplification that the mixed signal has a component at DC and at twice the demodulation frequency. This components at 2 times the demodulation frequency (called the 2 omega component) is effectively removed by regular low pass filtering. By selecting filters with small bandwidth and faster roll-offs, the 2 omega component can easily be attenuated by 100 dB or more. The problem arises at low demodulation frequencies, because this forces the user to select long integration times (e.g. >60 ms for a demodulation frequency of 20 Hz) in order to achieve the same level of 2 omega attenuation.

It is also the case that lock-in amplifiers will modulate DC offsets and non-linearities at the signal input with the demodulation frequency, resulting in a signal at the demodulation frequency (called omega component). This component is also effectively removed by the regular low-pass filters at frequencies higher 1 kHz.

At low demodulation frequencies, and especially for applications with demodulation frequencies close to the filter bandwidth, the omega and 2 omega component affects the measurement result. Sinc filtering permits to considerably attenuate the ω and 2 ω components. Technically the Sinc filter is a comb filter with notches at integer multiples of the demodulation frequency (ω, 2 ω, 3 ω, etc.). It removes the omega component with a suppression factor of around 80 dB. The amount of 2 omega component that gets removed depends on the input signal. It can vary from entirely (e.g. 80 dB) to slightly (e.g. 5 dB). This variation is not due to the Sinc filter performance but depends on the bandwidth of the input signal.

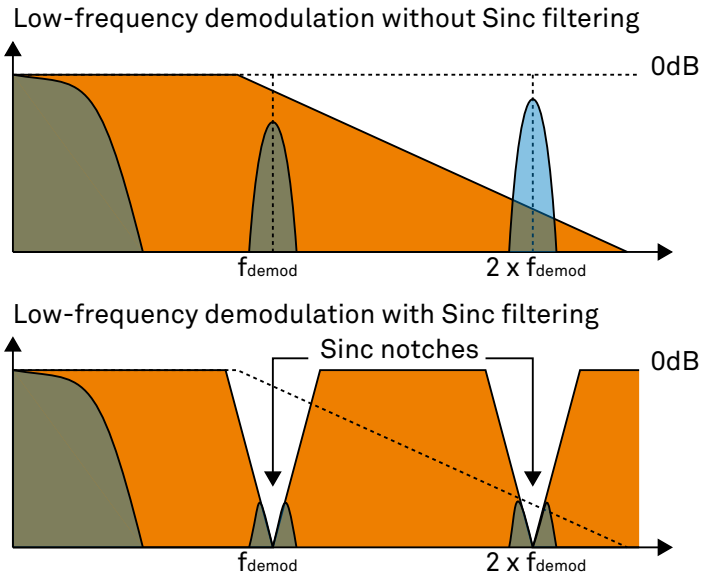


Figure 10.7. Effect of Sinc filtering

Table 10.3. Artifacts in the demodulation signal

Input signal	Demodulation result before low-pass filter	Result
Signal at omega	DC component	Amplitude and phase information (wanted signal)
	2 ω component	Unwanted component (can additionally be attenuated by Sinc filter)

Input signal	Demodulation result before low-pass filter	Result
DC offset	ω component	Unwanted component (can additionally be attenuated by Sinc filter)

A 30 Hz, 100 mV signal is generated and demodulated with 100 Hz bandwidth and 8th order. In addition 0.1 V offset is added to the signal so that we get a significant omega component. The offset voltage is generated by one of the auxiliary outputs instrument and added to the signal using the Add input of the signal output (see Figure 10.8). Then the Sinc filter is enabled, and Figure 10.9 shows in the Zoom FFT how the sinc options dampens both the ω and 2 ω components by about 100 dB.

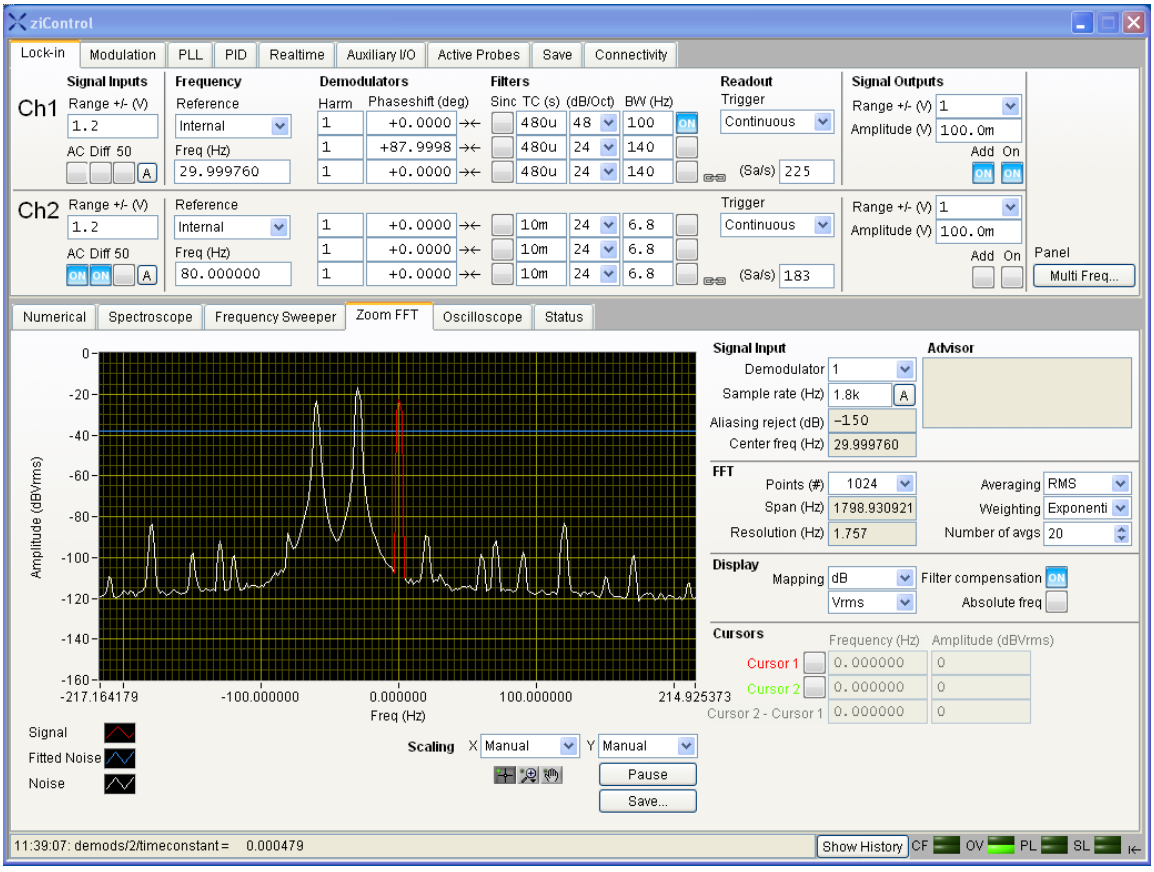


Figure 10.8. Example 30 Hz signal with large bandwidth demodulation without Sinc

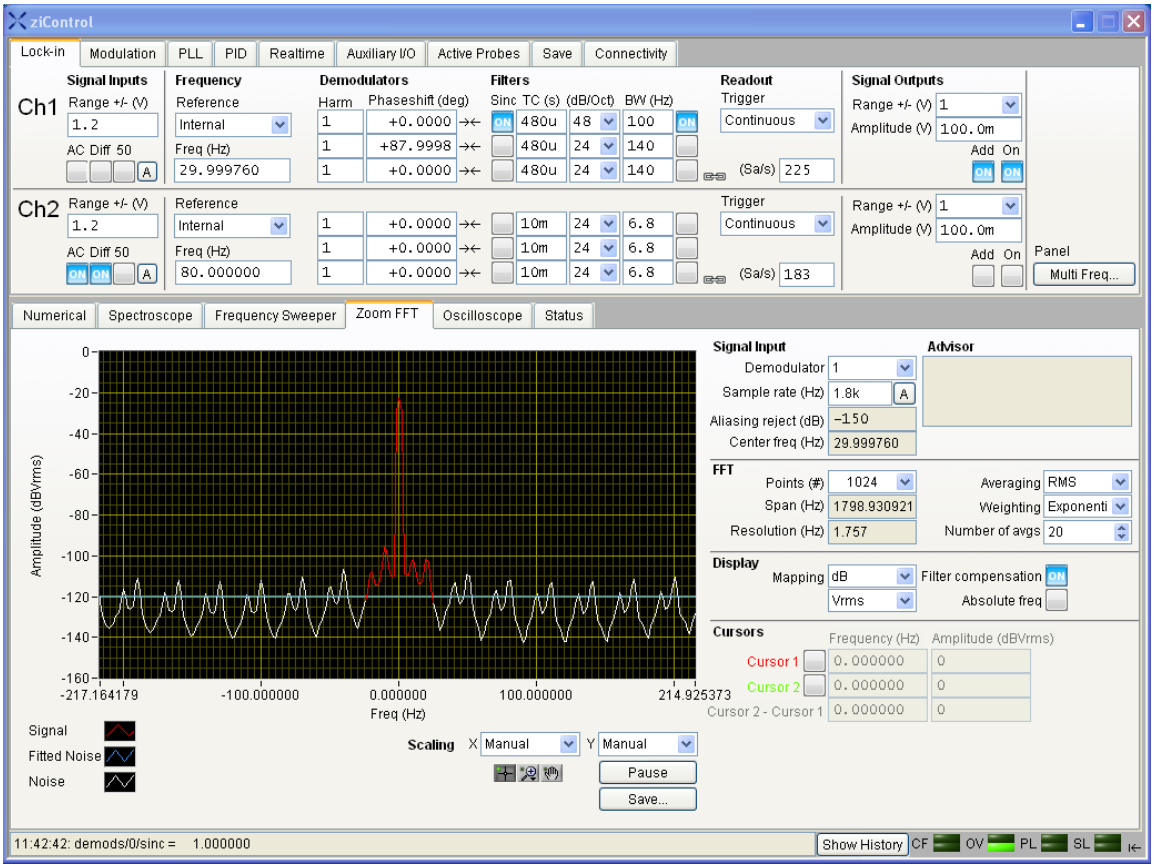


Figure 10.9. Example 30 Hz signal with large bandwidth demodulation with Sinc

Note

In order to put the notches of the digital filter to ω and 2 ω, the sampling rate of the filter would have to be precisely adjusted to the signal frequency. As this is technically not feasible, the generated signal frequency is adjusted instead. In this case the generated signal and the reference frequency are shifted to 29.99976 Hz to achieve maximum suppression.

10.6. Zoom FFT

The zoomFFT allows the user to analyze the spectral component of the input signal at and around a particular frequency by zooming in on a narrow frequency portion of the spectrum. This is done by performing a Fourier transform of the demodulated in-phase and quadrature (X and Y) components or more precisely, on the complex quantity $X + \mathrm{jY}$, where j is the imaginary unit.

As in normal FFT, where the sampling rate determines the frequency span and the ratio between sampling rate and sample number determines the minimum frequency resolution, the zoomFFT relies on the lock-in demodulation process to down shift the signal frequency, thereby allowing one to use both a much lower sampling rate and sample number to achieve the same frequency resolution. The advantage of the zoomFFT is in the size of the input signal: FFT needs to perform the computation on a long trace, only to retain the portion of the frequency spectrum of interest and to discard most of it. Typically, to achieve a 1 Hz frequency resolution at 1 MHz, FFT would require to collect (and compute) approximately 10^6 points, while zoomFFT only on 10^3. (Of course the high rate sampling is done by the lock-in during the demodulation stage, so the zoomFFT still needs to implicitly rely on a fast digitizer.)

In order to illustrate why this is so and what benefits this measurement tool brings to the user, it is useful to remind that at the end of the demodulation of the input signal $V_S(t) = A_S \cos\left(\omega_S t + \varphi_S\right)$, the output signal is $X + \mathrm{jY} = F(\omega_S - \omega_R)\frac{A_S}{\sqrt{2}} e^{j[(\omega_S - \omega_R)t + \varphi_S]}$ where $F(\omega)$ is the frequency response of the filters.

Since the demodulated signal has only one component at frequency $\omega_S - \omega_R$, its power spectrum (Fourier transform modulus squared) presents a peak of height $\frac{|A_S|^2}{2}|F(\omega_R - \omega_S)|^2$ at $\omega_S - \omega_R$: this tells us the spectral power distribution of the input signal at frequencies close to ω_R within the demodulation bandwidth set by the filters $F(\omega)$.

Note that:

- the ability of distinguish between positive and negative frequencies works only if the Fourier transform is done on $X + \mathrm{jY}$. Had we taken X for instance, the positive and negative frequencies of its power spectrum would be equal (the symmetry relation $G(-\omega) = G^*(\omega)$ holds for the Fourier transform $G(\omega)$ of a real function $g(t)$) and two peaks would appear at $\pm|\omega_S - \omega_R|$;
- One could extract the amplitude of the input signal by diving the power spectrum by $|F(\omega)|^2$, the operation being limited by the numerical precision. (This is implemented in ziControl and is activated by the Filter Compensation button: with the Filter Compensation enabled, the background noise appears white; without, the effect of the filter roll-off becomes visible.)

The case of an input signal containing a single frequency component can be generalized to the case of multiple frequencies (multiple signals or noise contributions) and the power spectrum would display all the frequency components weighted by the filter transfer function (or normalized if the Filter Compensation is enabled).

Also the digital nature of the HF2 does not modify the gist of the results reached above. When dealing with discrete time signal processing, one has to be careful about aliasing which originates when the signal frequencies higher than the sampling rate Ω are not sufficiently suppressed. Remember that Ω is the user settable readout rate, not the 210 MSa/s sampling rate of the HF2 input. Since the discrete time Fourier transform extends between $-\frac{\Omega}{2}$ and $+\frac{\Omega}{2}$, the user has to

make sure that at $\pm\frac{\Omega}{2}$ the filters provide the desired attenuation: this can be done either by increasing the sampling rate or resolving to measure a smaller frequency spectrum (i.e. with a smaller bandwidth).

Similarly to the continuous case, in which the length of the recorded trace determines the maximum frequency resolution ($\frac{2\pi}{T}$ if T is the recorded length), the resolution of the zoomFFT can be increased by increasing the recorded data points. If N data points are collected at a sampling rate Ω, the discrete Fourier transform has a frequency resolution of $\frac{\Omega}{N}$.

Chapter 11. HF2CA Current Amplifier Datasheet

This chapter contains the datasheet of the HF2CA Current Amplifier which is a pre-amplifier dedicated to the HF2 Series instruments. This datasheet is distributed only as part of the HF2 User Manual, and therefore not available separately.

The content of the chapter starts with the list of key features of the pre-amplifier, and continues with sections including the specifications, the detailed functional description, several possible applications, and finally an extended recommendation for 3rd party cables and connectors.

11.1. Key Features

- Current amplifier for high capacitive loads - shunt resistor based
- Voltage amplifier with selectable gain 1 or 10
- Input impedance switchable between 10 V/A and 1 MV/A
- Bandwidth from DC up to 100 MHz
- 2 differential amplification channels with switchable AC/DC coupling
- Adjustable output gain of 1 or 10
- Very low noise and small input leakage
- Single connector for power supply and control

The HF2CA current amplifier converts a differential input current to a differential output voltage in a wide frequency range. This device functions as an active probe and is conveniently placed close to the measurement setup. It supports applications with high capacitive loads such as dielectric impedance spectroscopy. When no shunt resistor is selected, the current amplifier works as a voltage amplifier. The careful design of the HF2CA insures stable operation over the entire frequency range.

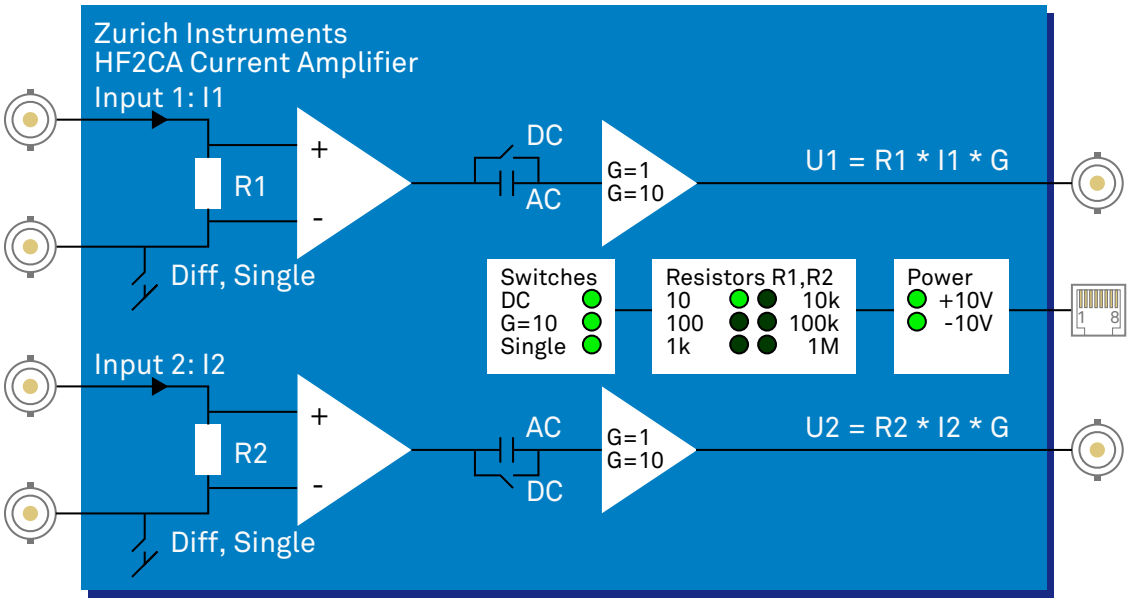


Figure 11.1. HF2CA functional overview

11.2. Specifications

Unless otherwise stated, all specifications apply after 30 minutes of device warming up.

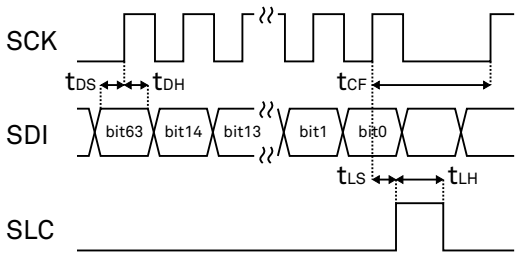
Table 11.1. General

Parameter	Description
dimensions	100 x 60 x 25 mm
weight	0.4 kg
storage temperature	-20 °C to 65 °C
operating temperature	5 °C to 40 °C
specification temperature	25 °C
specification supply voltage	12 V
connectors	4 SMB inputs, 2 SMB outputs, 1 RJ45 (no Ethernet)

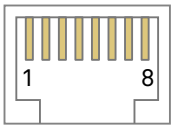
Table 11.2. Specifications

Parameter	min	typ	max
positive supply voltage VDD+	12 V	15 V	20 V
negative supply voltage VDD-	-20 V	-15 V	-12 V
supply current	60 mA	80 mA	120 mA
frequency response			
frequency range	DC	-	100 MHz
frequency range (AC coupled)	100 Hz	-	100 MHz
small signal bandwidth / 3dB cut-off (0.1 V_{PP} input, 50 pF output load, gain 1)	100 MHz	-	-
small signal bandwidth / 3dB cut-off (0.1 V_{PP} input, 50 pF output load, gain 10)	25 MHz	-	-
large signal bandwidth / 3dB cut-off (1 V_{PP}, 50 pF)	40 MHz	-	-
input			
input voltage noise (10 kHz)	-	7 nV/√Hz	-
input voltage noise (10 MHz)	-	6 nV/√Hz	-
input bias current	-	2 pA	10 pA
transimpedance gain (equivalent to input impedance)	10 V/A	-	1 MV/A
transimpedance gain accuracy (G=1)	-	±0.1 %	-
transimpedance gain accuracy (G=10)	-	±1 %	-
input offset voltage	-	-	1 mV
common-mode offset range	-10 V	-	7.5 V
output			
output voltage gain	1	-	10
control interface			
input high level	2.0 V	-	5 V
input low level	0 V	-	0.8 V

Parameter	min	typ	max
all transitions on SDI, SDO, SCK, SLC	-	-	1 µs
SCK clock period	10 µs	-	-
SDI data to clock setup t_{DS}	2 µs	-	-
SDI data hold from clock t_{DH}	1 µs	-	-
SLC clock to latch setup t_{LS}	1 µs	-	-
SLC latch hold t_{LH}	10 µs	-	20 µs
SCK clock free time t_{CF}	20 µs	-	-



ZCtrl: Control & Power Connector RJ45 (no Ethernet!)



Pin Assignment:
1 VDD+
2 GND
3 GND
4 VDD-
5 SLC
6 SCK
7 SDI
8 SDO

Figure 11.2. Digital control timing

Some parameters depend on the transimpedance gain settings. The following table provides an overview. The values in this table are typical values, they depend on the source capacitance, on the input signal swing, and as well as on the capacitive load on the output of the amplifier.

Table 11.3. Gain dependent parameters

Input impedance setting	Bandwidth / 3dB cut-off frequency	Maximum input current range	Maximum input current noise
10 V/A	100 MHz	±160 mA	400 pA/√Hz
100 V/A	50 MHz	±16 mA	42 pA/√Hz
1 kV/A	5 MHz	±1.6 mA	5.6 pA/√Hz
10 kV/A	500 kHz	±160 µA	1.3 pA/√Hz
100 kV/A	50 kHz	±16 µA	400 fA/√Hz
1 MV/A	5 kHz	±1.6 µA	128 fA/√Hz

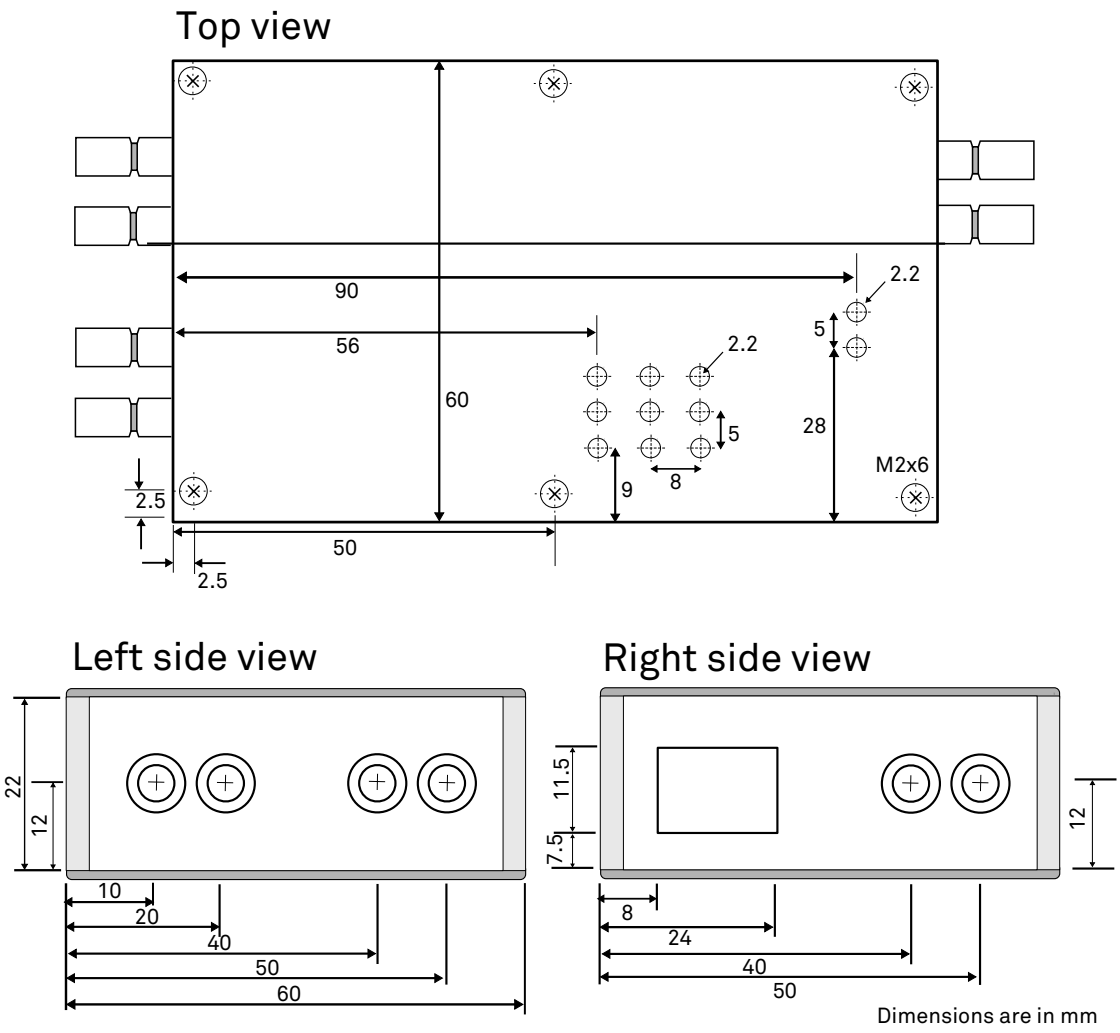


Figure 11.3. Casing dimensions of the HF2CA

11.3. Functional Description

The HF2CA external amplifier can be placed close to the signal source whereas the HF2 Instrument can be several meters away. Such a setup significantly improves the measurement quality due to less parasitics effects and to smaller interferences.

The two signal channels of the HF2CA can be used as separate amplification channels, or alternatively, in differential mode connected to the differential input of the HF2 Instrument.

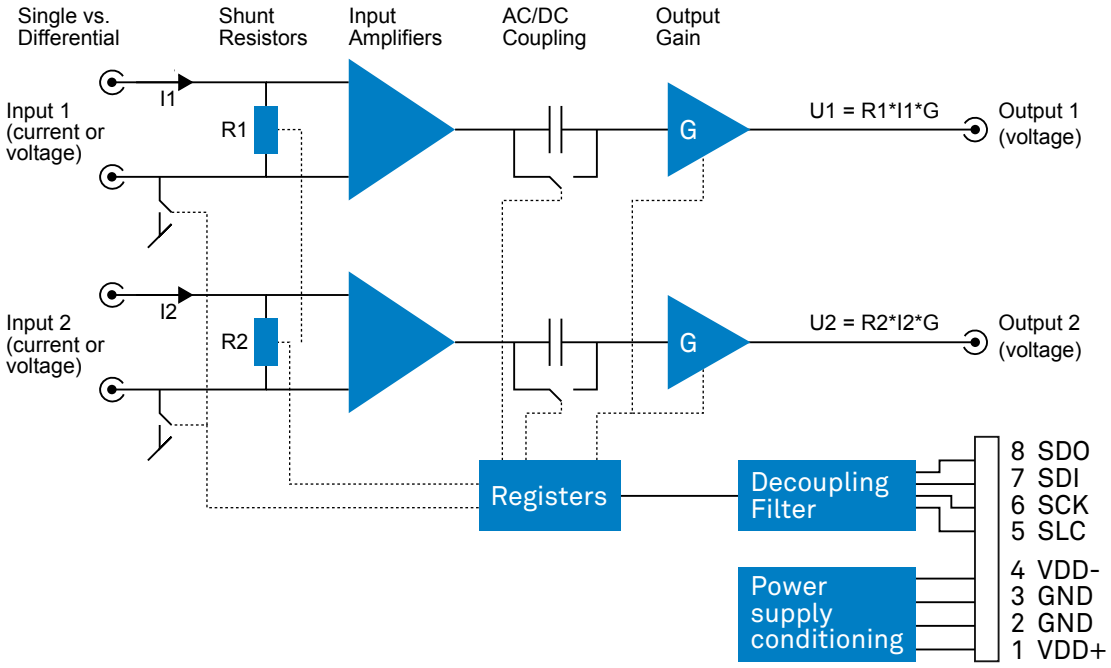


Figure 11.4. Detailed block diagram

11.3.1. Input and Output

Shunt resistors: HF2CA measures the current between the positive and the negative input terminal by measuring the voltage drop across a resistor, that is shunted between the inputs (see Figure 11.4) . The supported resistor values are given in Table 11.3 . It is also possible to turn-off all internal resistors and to support any custom resistor that is externally connected (see Section 11.4.4). In principle it is also possible to switch more than one resistor, but this is not supported by the HF2 Software. When all resistors are turned-off, then the HF2CA becomes a voltage amplifier with selectable gains 1 and 10.

JFET input amplifiers: the HF2CA is based on JFET input amplifiers that provide very low-noise over a wide frequency range. Additionally, the ultra-low input bias current of typically 2 pA allows for precise current measurements at small signal amplitudes. The input voltage range of the JFET input amplifiers is -10 V to 7.5 V for each input which is also the common mode offset range.

Single vs. differential mode: a selectable switch to amplifier ground allows the user to earth the negative terminal of each input and to operate in single-ended mode without needing external circuits. Alternatively, when leaving the ground switches open, it is possible to use a differential input signal or to connect the negative terminals to local ground externally.

AC vs. DC mode: a selectable switch after the input amplifiers allows the user to measure DC or close to DC signals, or when this is not required, to select AC coupling with a cut-off frequency at 100 Hz and eliminate potential 50/60 Hz noise from the measured signal.

11.3.2. Power Supply and Remote Control

The HF2CA is designed for use with the HF2 Series with its differential signal for improved signal-to-noise, and a single cable that provides power and control signals. A straight-through (as opposed to cross-over) Ethernet cable must be used. The cable carries the following signals:

- **Power**: positive and negative supply, ground
- **Digital control**: SDI digital input signal to control the preamplifier settings, SDO output signal for device detection (details of function not disclosed to users), SCK clock signal, and SLC latch signal. SDI, SCK and SLC are used to program the shift registers on the amplifier and thereby adjust the correct settings. The setting bits are given in Table 11.4 . The timing diagram of the digital interface is given in Figure 11.2 . The MSB of the register settings is shifted in first.

Table 11.4. HF2CA register settings

Register bit	Name	Description
15 to 10	-	unused
9	gain	0: set output gain to 1
		1: set output gain to 10
8	dcswitch2	0: set AC coupling for input 2
		1: set DC coupling for input 2
7	dcswitch1	0: set AC coupling for input 1
		1: set DC coupling for input 1
6	singleswitch	0: set differential operation
		1: set single-ended operation
5	res1m	1: set resistor 1 MV/A
4	res100k	1: set resistor 100 kV/A
3	res10k	1: set resistor 10 kV/A
2	res1k	1: set resistor 1 kV/A
1	res100	1: set resistor 100 V/A
0	res10	1: set resistor 10 V/A

11.4. Applications

- Impedance spectroscopy
- Large capacitive loads
- Wheatstone-bridge configuration
- Preamplifier for HF2IS impedance spectroscope and HF2LI lock-in amplifier

11.4.1. Differential Current Measurement with Common-mode Offset

The resistors at the input of the amplifier can be inserted in a current path as shown in the figures. With this, fast current transients can measured at large common-mode voltages, which are in the range from -10V to 7.5V are possible. This is used in, e.g., high-energy physics to record the radiation-induced current in a photo diode.

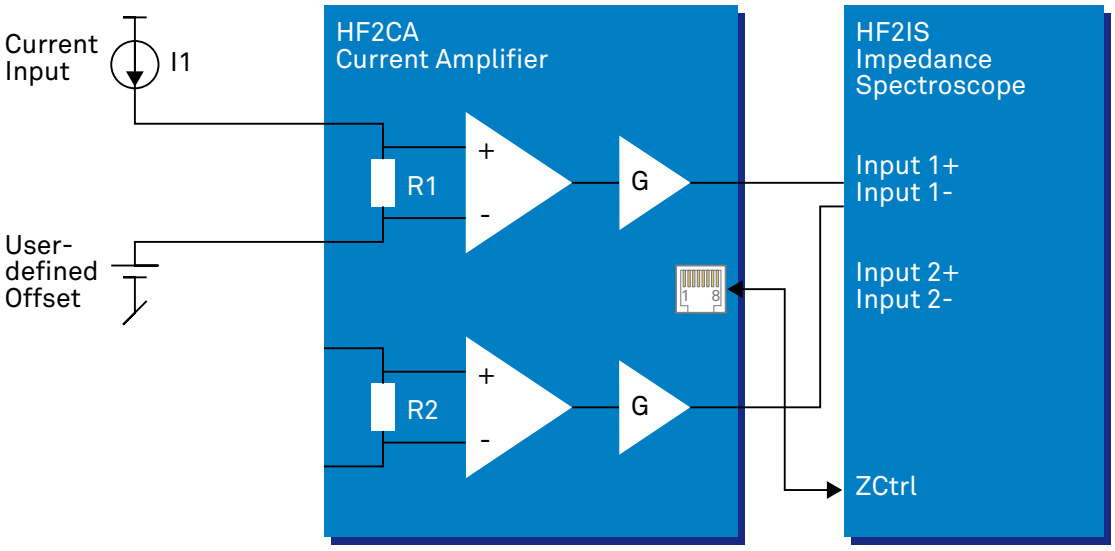


Figure 11.5. HF2CA differential current measurement

11.4.2. Multi-frequency Impedance Spectroscopy

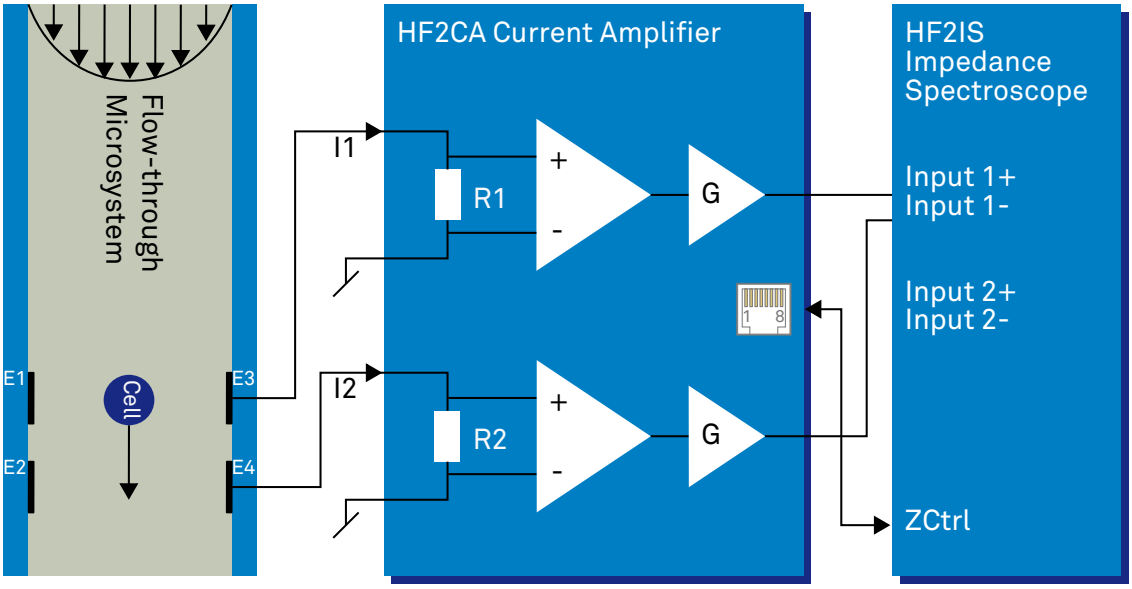


Figure 11.6. HF2CA impedance spectroscopy

The HF2CA in combination with the HF2IS impedance spectrometer is the solution to measure impedance in, for example, flow-through microsystems. The challenge here is to measure the channel impedance at high frequencies (>10 MHz). The large capacitance occurring at electrode electrolyte interfaces can lead to stability issues in a transimpedance amplifier. A solution is to use the electrodes in a Wheatstone bridge configuration with shunt resistors. The HF2CA offers this solution.

As shown in the figure, electrodes are place on the channel walls of a microfluidic channel (width in the order of 20 to 50 µm). Electrodes E1 and E2 are stimulated with a sinusoidal voltage, the electrodes E3 and E4 are connected to the positive amplifier inputs and thus shunted to GND via resistors R1 and R2. The resulting voltage drops across R1 and R2 are given by the channel impedance. This impedance varies when a particle or a living cell passes the electrode area. An analysis at multiple frequencies at the same time (which is supported by the HF2IS and the HF2CA) allows for concurrently analysing cell size and dielectric properties. With this information biologists, e.g. sort their cells and detect cell viability or health.

11.4.3. Impedance Measurement

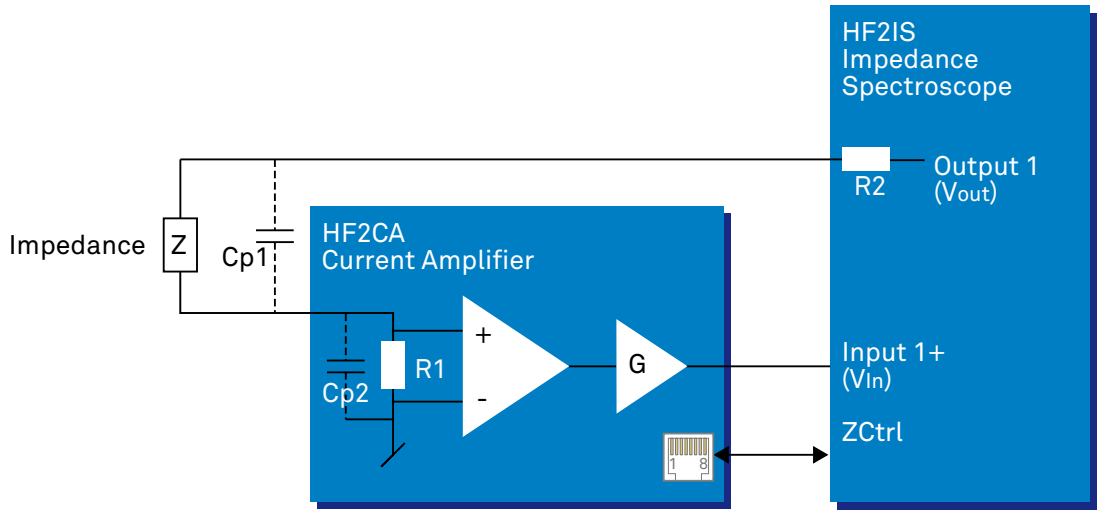


Figure 11.7. Measure an impedance using the HF2CA

The HF2 in conjunction with the HF2CA can be used to measure impedances at various frequencies. The connection diagram is shown in the figure above. The impedance of interest, Z, is connected to the input resistor in the HF2CA preamplifier. For optimal signal-to-noise, the input resistor, R1, is set to a value close to the impedance Z. The HF2 generates an output signal of amplitude V_{Out} and the output signal from the preamplifier is connected to the positive input (Input +) of the HF2, which is here called V_{In}. With this setup, the impedance Z can be calculated using the following equation:

$Z = R\,(V_{Out}-V_{In})/V_{In}$.

Here we neglected the output resistance, R2, of the HF2 device. The is valid as long as Z>>R1=50 Ohm. Furthermore, at high frequencies the parasitic capacitances Cp1 and Cp2 will have to be included in the calculation. At even higher frequencies, Cp1 and Cp2 will be dominant.

11.4.4. Custom Input Impedance

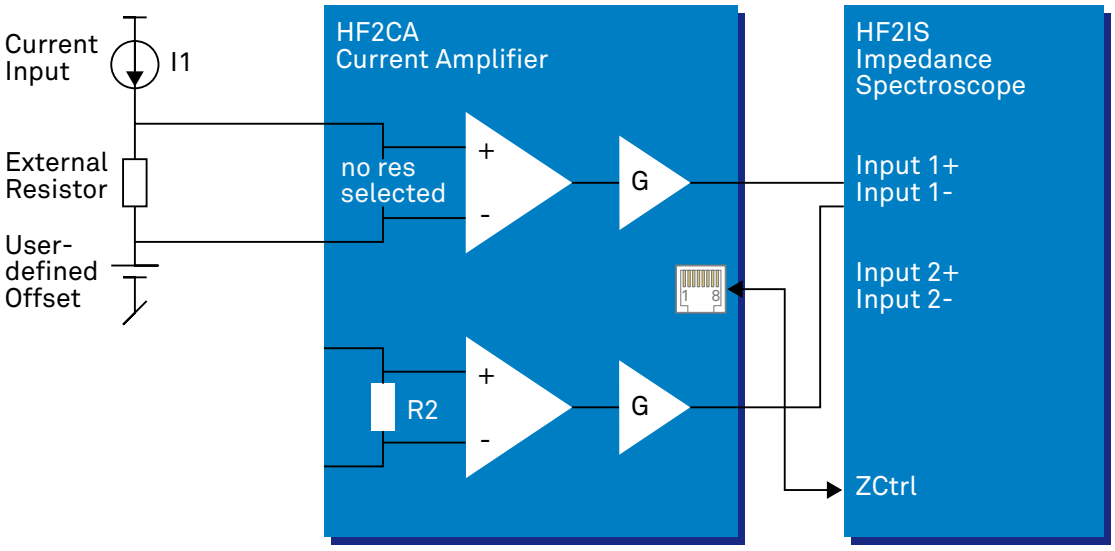


Figure 11.8. HF2CA custom input impedance

Sometimes it is useful to choose a special resistance value in order to optimize the signal to noise by, e.g. impedance matching. In this case, an external resistor can be used instead of using the standard values inside the preamplifier. All internal resistors need to be disconnected in this case, which can be done using the standard preamplifier user interface.

11.5. Cable Recommendation

Table 11.5. HF2CA cable recommendation

Function	Connector / cable type	Vendor / part number
SMB to BNC connection		
SMB to BNC cable	BNC jack to SMB plug	Farnell / Newark 1351896
SMB to BNC adapter	BNC jack to SMB plug	Digikey ACX1386-ND
	BNC jack to SMB jack	Farnell / Newark 4195930
Custom access or cable assembly		
Cable	Cable type RG-174	Digikey A307-100-ND
		Farnell / Newark 1387745
SMB to cable	SMB plug to RG-174 cable	Tyco Electronics 413985-1
		Digikey A4026-ND
		Farnell / Newark 2141206
BNC to cable	BNC plug to RG-174 cable	Tyco Electronics 1-5227079-6
		Digikey A32212-ND
		Farnell / Newark 1831701

Chapter 12. HF2TA Current Amplifier Datasheet

This chapter contains the datasheet of the HF2TA Current Amplifier which is a pre-amplifier dedicated to the HF2 Series instruments. This datasheet is distributed only as part of the HF2 User Manual, and therefore not available separately.

The content of the chapter starts with the list of key features of the pre-amplifier, and continues with sections including the specifications, the detailed functional description, several possible applications, information how to test the specified performance, and finally an extended recommendation for 3rd party cables and connectors.

12.1. Key Features

- 50 MHz operation range
- 2 independent amplification channels with selectable AC/DC coupling
- Wide range of current gain settings (100 V/A to 100 MV/A)
- Impedance measurements from 1 μΩ to 100 MΩ
- Input offset voltage adjustment
- Voltage output amplifier with selectable gain 1 or 10
- Very low noise and low input leakage
- Single connector for power supply and control

The HF2TA current amplifier converts 2 input currents to output voltages in a frequency range up to 50 MHz. This device is an active probe which can be conveniently placed close to the measurement setup. It supports most applications where a current must be converted to a voltage. The advanced design of the HF2TA ensures stability and a smooth operation over the entire frequency range. The HF2TA transimpedance current amplifier with the HF2 Series signal analyzers allows for very high performance measurements and insensitivity to interferences thanks to reduced parasitics.

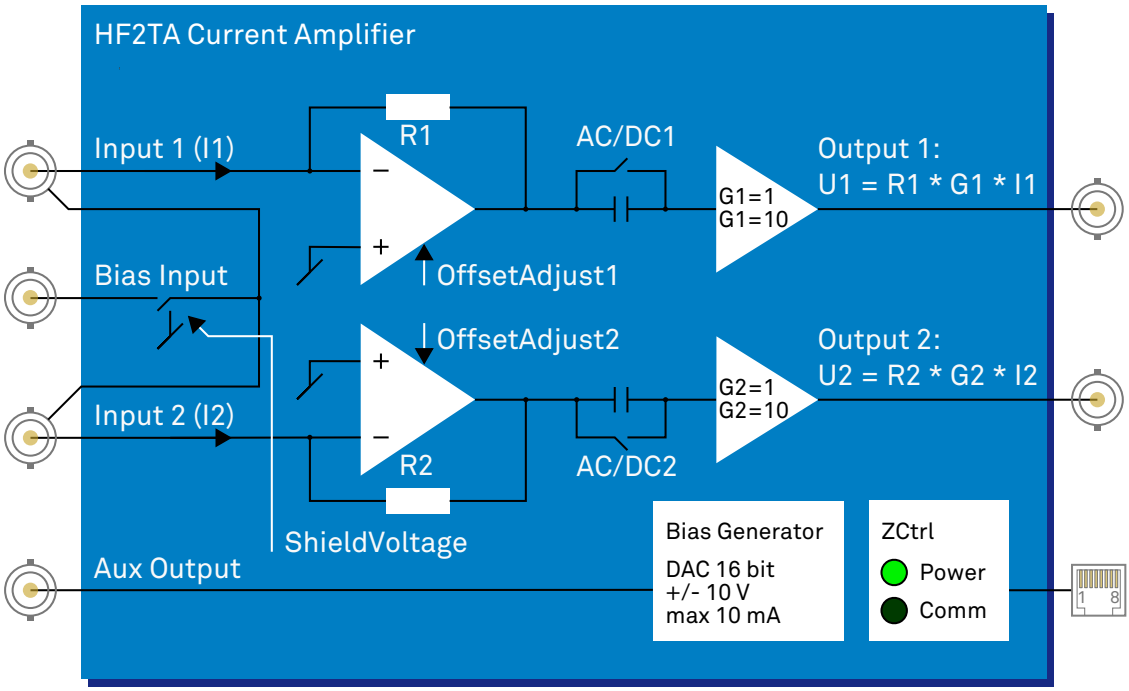


Figure 12.1. HF2TA functional overview

12.2. Specifications

Unless otherwise stated, all specifications apply after 30 minutes of device warming up.

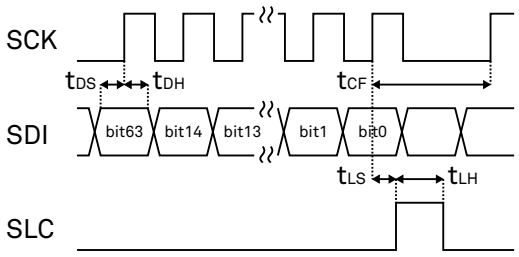
Table 12.1. General

parameter	description
dimensions	101 x 78 x 23 mm
weight	0.4 kg
storage temperature	-20 °C to 65 °C
operating temperature	5 °C to 40 °C
specification temperature	25 °C
connectors	3 SMA inputs female, 3 SMA outputs female, 1 RJ45 (no Ethernet)

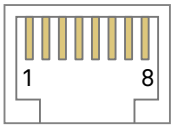
Table 12.2. Specifications

parameter	min	typ	max
positive supply voltage VDD+	12 V	13 V	15 V
negative supply voltage VDD-	-15 V	-13 V	-12 V
supply current	50 mA	60 mA	100 mA
frequency response			
frequency range	DC	-	50 MHz
frequency range (AC coupled)	10 Hz	-	50 MHz
small signal bandwidth / 3dB cut-off (0.1 V_{PP} input, 50 pF output load, gain 1)	-	-	50 MHz
small signal bandwidth / 3dB cut-off (0.1 V_{PP} input, 50 pF output load, gain 10)	-	-	50 MHz
large signal bandwidth / 3dB cut-off (1 V_{PP}, 50 pF)	-	-	40 MHz
input			
input current range	depends on R1, R2, G1, G2 settings		
input current noise	depends on R1, R2, G1, G2 settings		
input voltage noise (10 kHz)	-	7 nV/√Hz	-
input voltage noise (10 MHz)	-	5 nV/√Hz	-
input leakage current	-	2 pA	20 pA
input voltage offset compensation range	-10 mV	-	10 mV
input impedance range (Z // 15 pF)	50 Ω	-	70 kΩ
input bias voltage range	-10 V	-	10 V
input signal level (damage threshold)	-5 V	-	5 V
output			
output voltage gain (G1,G2)	1	-	10
transimpedance gain (R1,R2)	100 V/A	-	100 MV/A

parameter	min	typ	max
transimpedance gain accuracy (R1,R2)	-	±1 %	-
digital control interface timing			
input high level	2.2 V	-	5 V
input low level	0 V	-	0.8 V
all transitions on SDI, SDO, SCK, SLC	-	-	1 µs
SCK clock period	10 µs	-	-
SDI data to clock setup t_{DS}	2 µs	-	-
SDI data hold from clock t_{DH}	1 µs	-	-
SLC clock to latch setup t_{LS}	1 µs	-	-
SLC latch hold t_{LH}	10 µs	-	20 µs
SCK clock free time t_{CF}	20 µs	-	-



ZCtrl: Control & Power Connector RJ45 (no Ethernet!)



Pin Assignment:
1 VDD+
2 GND
3 GND
4 VDD-
5 SLC
6 SCK
7 SDI
8 SDO

Figure 12.2. Digital control interface timing

Some parameters depend on the transimpedance gain settings. The following table provides an overview. The values in this table are typical values, they depend on the source capacitance, on the input signal swing, and also on the capacitive load on the output of the amplifier.

Table 12.3. Gain dependent parameters 1

input impedance setting	bandwidth / 3dB cut-off	maximum input current range (G=1)	maximum input current range (G=10)
100 V/A	50 MHz	±10 mA	±1 mA
1 k V/A	50 MHz	±1 mA	±100 µA
10 kV/A	8 MHz	±100 µA	±10 µA
100 kV/A	1.5 MHz	±10 µA	±1 µA
1 MV/A	250 kHz	±1 µA	±100 nA
10 MV/A	25 kHz	±100 nA	±10 nA
100 MV/A	12 kHz	±10 nA	±1 nA

Table 12.4. Gain dependent parameters 2

input impedance setting	input impedance	maximum input current noise	measured at
100 V/A	50 Ω	150 pA/√Hz	1 MHz
1 k V/A	50 Ω	15 pA/√Hz	1 MHz
10 kV/A	50 Ω	2 pA/√Hz	1 MHz

input impedance setting	input impedance	maximum input current noise	measured at
100 kV/A	100 Ω	500 fA/√Hz	100 kHz
1 MV/A	300 Ω	250 fA/√Hz	100 kHz
10 MV/A	1.6 kΩ	100 fA/√Hz	10 kHz
100 MV/A	70 kΩ	50 fA/√Hz	10 kHz

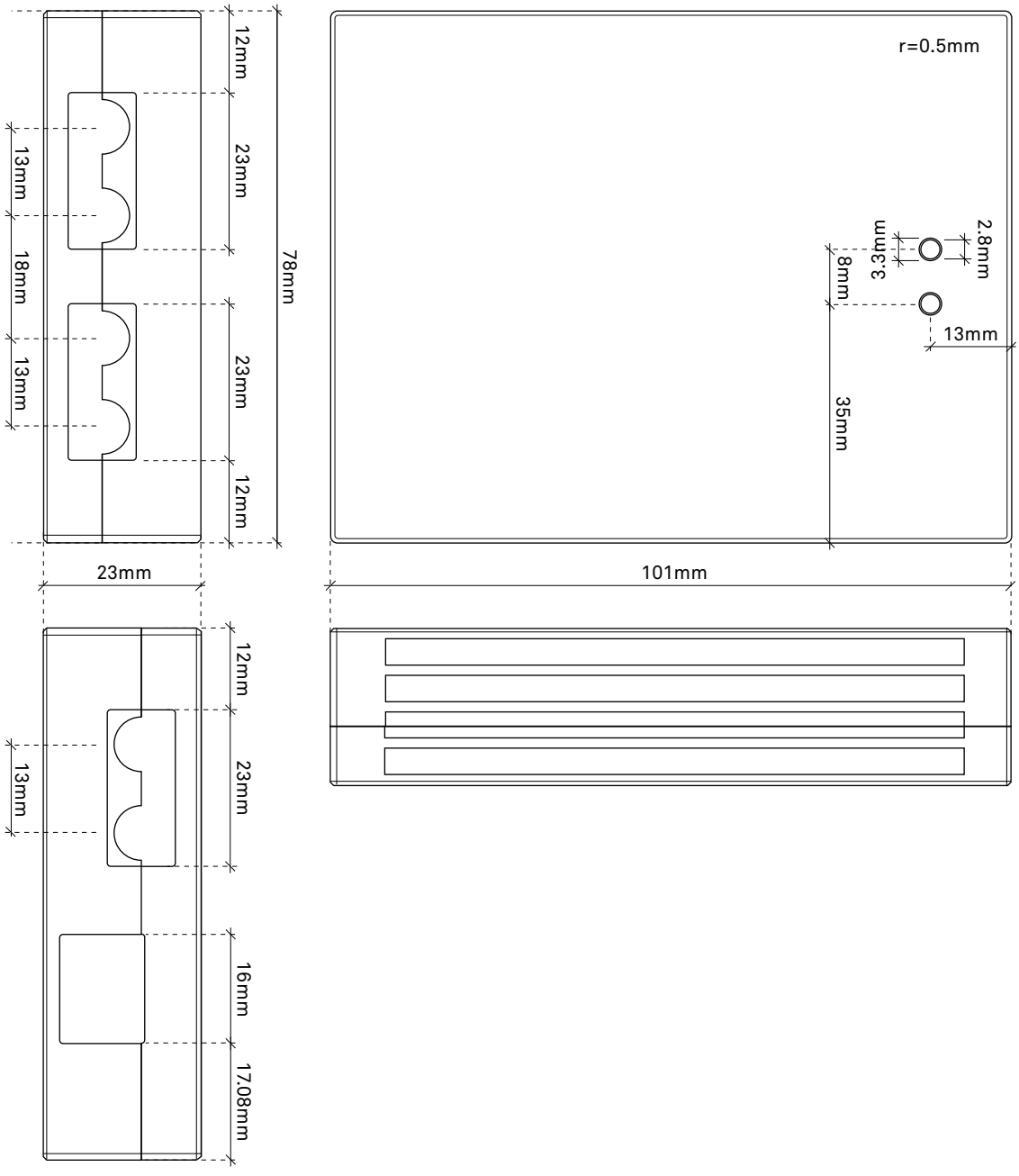


Figure 12.3. Casing dimensions of the HF2TA

12.3. Functional Description

The HF2TA is an external current preamplifier for the HF2 Series instruments from Zurich Instruments. The preamplifier can be placed close to the signal source, which significantly improves the measurement quality due to less parasitics effects and to smaller interferences.

The two signal channels of the HF2TA can be used as separate current amplification channels, or alternatively, in differential mode connected to the differential input of the HF2 Instrument. The channels settings can be set independently.

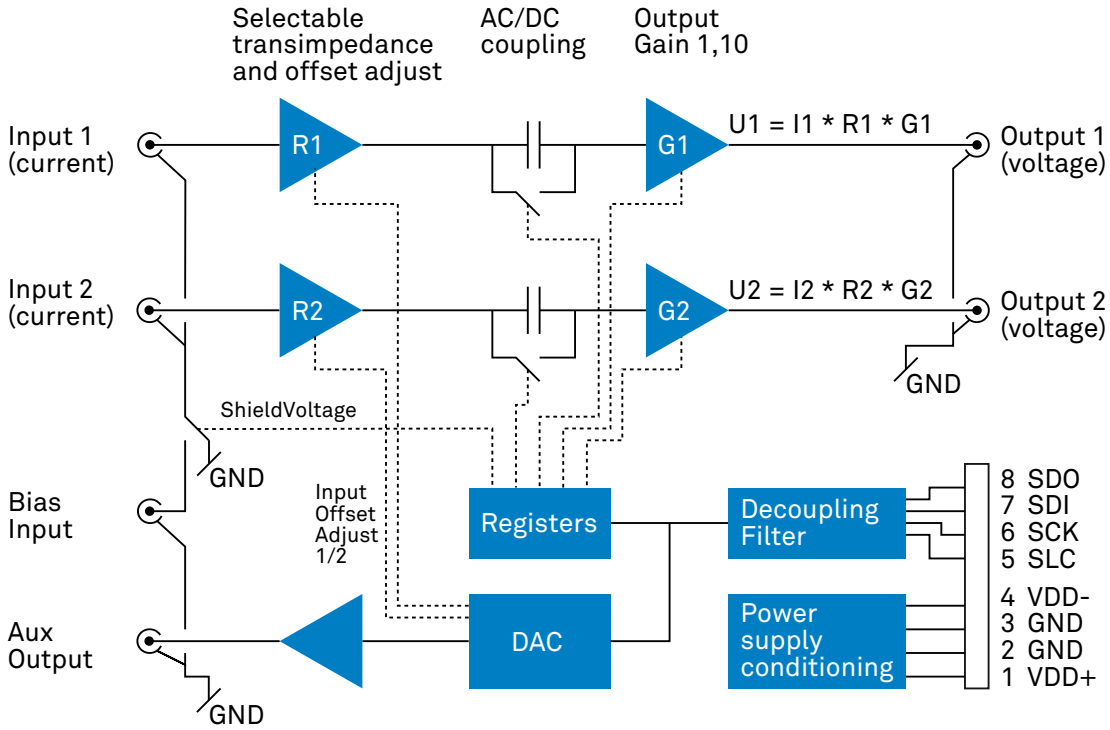


Figure 12.4. Detailed block diagram

12.3.1. Input and Output

Transimpedance stage: the HF2TA measures the current flowing at the two input terminals. The current amplifier uses a standard transimpedance stage to convert the current to a voltage output. The input terminal is matched to 50 Ohms to allow for proper impedance matching at high frequencies. At high current gains, or low input currents, respectively, the input terminal acts like a low-impedance virtual ground. The input impedance depends on the gain settings as described in the table above.

Voltage gain 1 or 10: the HF2TA offers a voltage gain of either 1 or 10 after the transimpedance amplifier. This allows to optimize the signal-to-noise at small amplitudes and high bandwidths. The transimpedance gain often has to be kept small in order to meet the required bandwidth. A voltage amplifier helps in this case to improve the measurement quality.

JFET input amplifiers: the HF2TA is based on JFET input amplifiers that provide very low-noise over a wide frequency range. Additionally, the ultra-low input bias current of typically 2 pA allows for precise current measurements at small signal amplitudes. The input voltage range of the JFET input amplifiers is -5 V to 2 V for each input which is also the common mode offset range.

Offset adjustment: the offset of the input amplifier can be manually compensated. For this purpose, disconnect any signal from the input of the current amplifier and measure the output

voltage. Change the offset voltage until the output is close to zero. All remaining offset should now come from other sources (like offset current or leakage from the device under test).

AC vs. DC mode: a selectable switch after the input amplifiers allows the user to measure DC signals, or when this is not required, to select AC coupling with a cut-off frequency at around 10 Hz to remove the DC offset. When working in AC, make sure that the first amplifier is not saturating. This can be checked by switching to DC and gain 1.

Bias output: the HF2TA comprises a general purpose low-noise analog output. This output can be used as a power supply for, e.g., photo diodes. The photo diode is connected to the auxiliary bias output and the virtual ground of the input, no additional power supply is needed.

Signal shield voltage: the bias input connector can be used to apply a bias voltage to the signal shield. This can be used, for instance, to power a remote sensor over the signal shield without introducing an additional ground loop. If this option is not used, the signal shield should be conveniently grounded with the control setting ShieldVoltage.

12.3.2. Power Supply and Remote Control

The HF2TA is designed for use with the HF2 Series devices. It has to be connected to the ZCtrl 1/2 connectors of the host device using a single Ethernet cable which provides both power and control signals. A standard straight-through (as opposed to cross-over) cable must be used. The cable carries the following signals:

- **Power**: positive and negative supply, ground
- **Digital control**: SDI digital input signal to control the preamplifier settings, SDO output signal for device detection (details of function not disclosed to users), SCK clock signal, and SLC latch signal. SDI, SCK and SLC are used to program the shift registers on the amplifier and the DAC and thereby adjust the correct settings.

12.4. Applications

- Low-noise and high-speed current amplification
- Photo diode preamplifier
- Impedance measurement
- Semiconductor testing
- Impedance spectroscopy

12.4.1. Recommended Settings

In order to get the maximum performance out of your HF2TA, the following guidelines should be followed.

- low and high input current measurement

  The HF2TA gain setting should be selected properly in the measurement path. The gain setting can be set according to Table 15.3. Gain dependent parameter 1 . As one can see, each input impedance and G setting has a maximum input current range specified. With each recommended input impedance and G setting, the maximum current will produce the maximum voltage swing of ±1V at the output of the HF2TA. At this level the input digitizer of the HF2 input channel will run close to its full dynamic range which results in the optimal SNR.

- low and high bandwidth measurement

  HF2TA is specified to work up to the 3dB bandwidth of 50MHz. Nevertheless, cares must be taken when selecting input impedance gain settings. Table 15.3. Gain dependent parameter 1 details as well the maximum 3dB signal bandwidth for each gain setting. For example, with an input current containing frequency components of less than 12kHz in frequency, the maximum transimpedance gain of 100MV/A can be selected. At 50MHz, only 100V/A of transimpedance gain is available. G=10 can also be selected as well if more gain is required at high input signal frequencies.

- minimize cross-talk and parasitics effects

  With the measured impedance placed closely to the input of the HF2TA and the HF2 device, four point measurement setup can help to minimize parasitic effect as well as the noise pickups from the cable. Furthermore, using shielded cable can greatly reduce the high frequency noise pickups from the surrounding environment.

- avoid HF2TA instability

  Since HF2TA is a negative feedback amplifier, its feedback loop stability can be sensitive to input capacitance, especially at low R settings. In order to avoid possible under-damped behaviour (i.e. oscillation) in the measurement, it is recommended to use as high as possible the selected transimpedance gain R when measuring a capacitive circuit. A short cable to the HF2TA input can also help to reduce the parasitic capacitance seen at the HF2TA input.

12.4.2. Photo Diode Amplifier with HF2LI

The HF2TA current amplifier is suited to read out the current from a photo diode. The following figures shows three possible ways to use the device. In the first option, the photo diode is grounded on one side and connected to the current amplifier on the other side. The recorded signal is amplified and sent to the HF2 Instrument.

The second option provides a solution when it is necessary to apply a bias voltage across the photo diode. For this purpose the bias output of the HF2TA can be used. Voltages in the range of +/-10V and currents up to 10mA can be delivered by this connector. Alternatively the bias can be provided by another voltage source.

The thirds option supports the drive of the photo diode by means of the shield of the signal cable. This shield can be conveniently driven by the HF2TA by shortcutting the bias output to the bias input. This option permits the user to connect the remote sensor with one single coaxial cable and while avoiding to introduce a ground loop in the system.

All HF2TA settings can be conveniently programmed inside the graphical user interface of the HF2 Instrument.

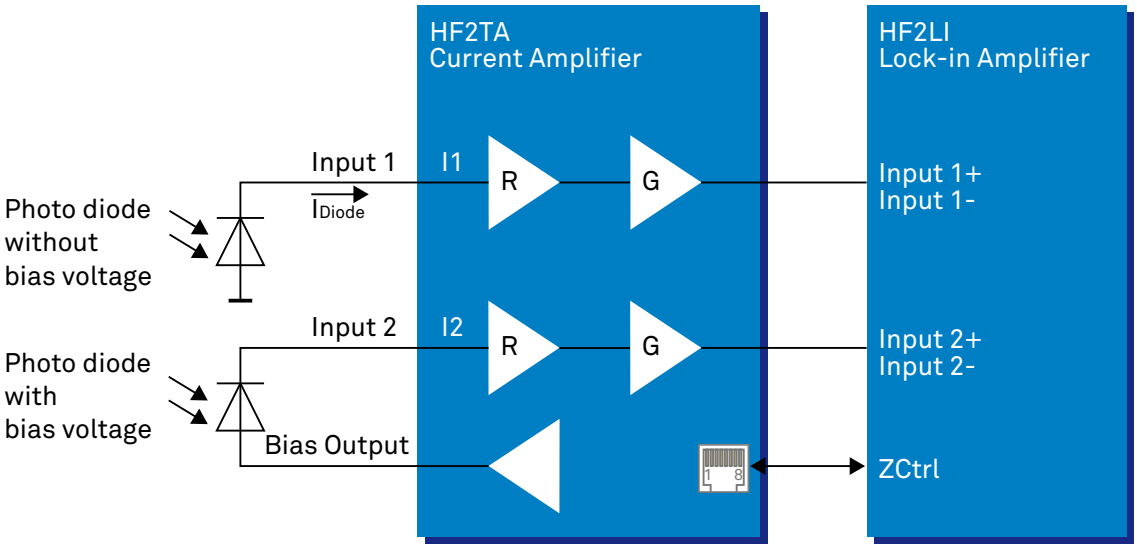


Figure 12.5. HF2TA photo diode amplifier

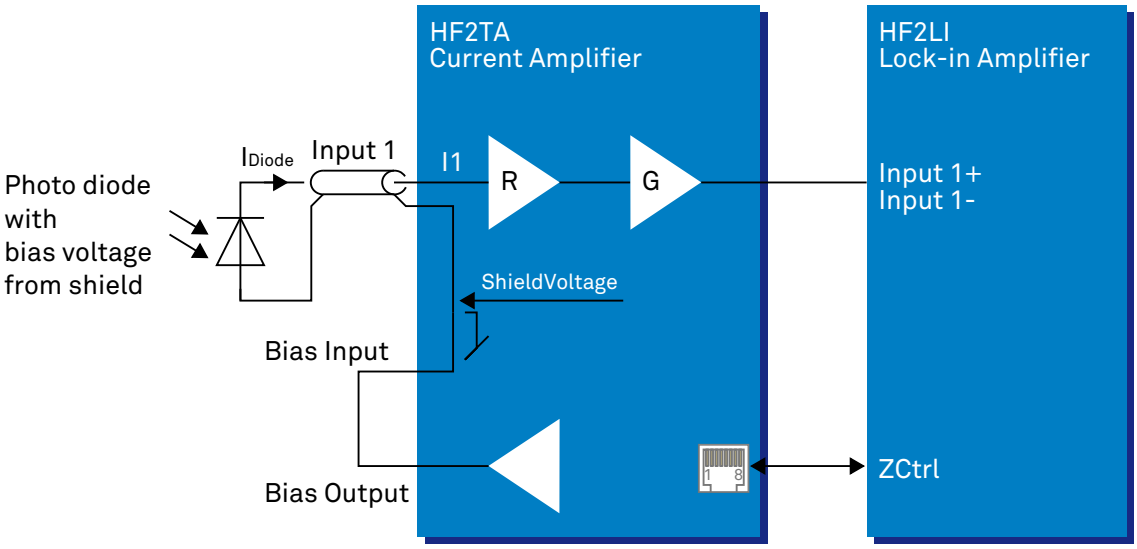


Figure 12.6. HF2TA photo diode amplifier with single coaxial cable

12.4.3. Impedance Measurement with HF2IS

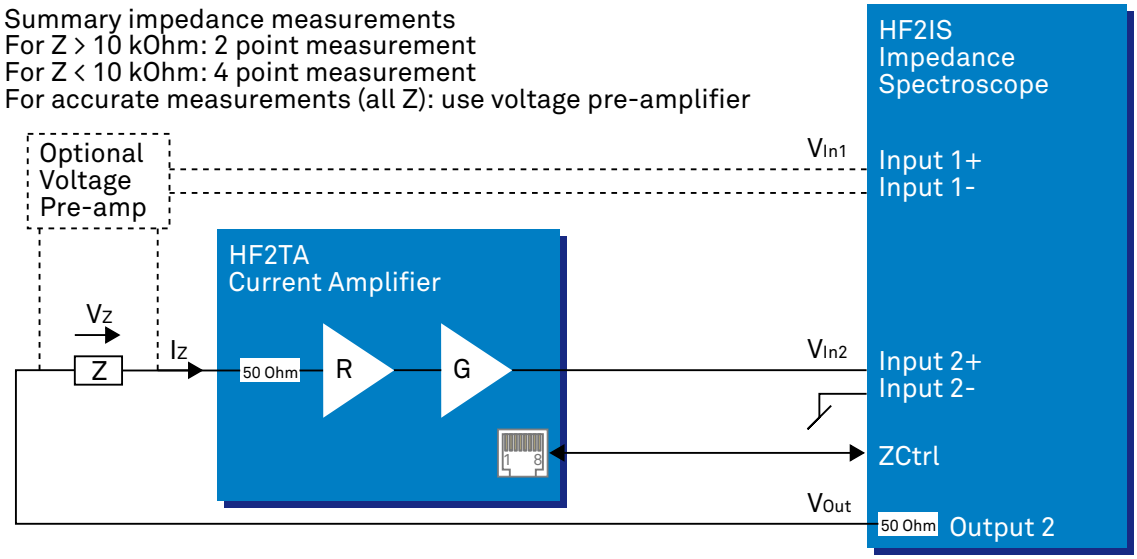


Figure 12.7. Measure an impedance using the HF2TA

The HF2TA current amplifier can be used in conjunction with the HF2IS instrument to measure impedances in a very wide range at frequencies up to 50 MHz. The connection diagram in the figure above shows how the impedance of interest Z is connected to the input of the HF2TA. For optimal amplification versus bandwidth setting, the table in the specification section may be consulted. Three cases and applications need to be distinguished.

- Measuring an impedance Z > 10 kΩ

  For large impedances it is possible to neglect the output resistance of the HF2IS Instrument and the input resistance of the preamplifier, thus the simple setup provides good accuracy. The HF2IS generates an output signal of amplitude V_{Out} and the output signal from the preamplifier is connected to the positive Input 2+ of the HF2, called V_{In}. With this setup, the impedance Z can be calculated using the following equation:

  $Z = R * G * V_{Out} / V_{In2}$

- Measuring an impedance Z < 10 kΩ

  For small impedances and higher precision a four point measurement setup is required. For accuracy in the range of 1%, the voltage V_Z can be measured directly by the second differential Input 1+ and Input 1- of the HF2. In this case it is important to select the high ohmic input impedance option (1 MΩ) as otherwise too much current is dissipated in the measurement instrument. Also the HF2 should be configured for differential measurement. The resulting impedance Z is calculated using the following equation:

  $Z = R * G * V_Z / V_{In2} = R * G * V_{In1} / V_{In2}$

- Measuring impedances with high accuracy (all values of Z)

  Four point measurement setup allows the most accurate measurement by taking into account simultaneously the current flowing through the measured impedance and the voltage drop caused by the current flow. For an accuracy better than 1%, it is recommended to use a voltage pre-amplifier with high-ohmic input stage to measure the voltage across the impedance $V_Z = V_{In1}$. Assuming Vin2, R and G are the output, the resistor setting and the gain of the HF2TA, respectively, the resulting equation to calculate the impedance will be similar to the previous case:

  $Z = R * G * V_Z / V_{In2} = R * G * V_{In1} / V_{In2}$ (assuming voltage pre-amp gain = 1)

A pictorial representation of how to set up the four-point measurement is shown below.

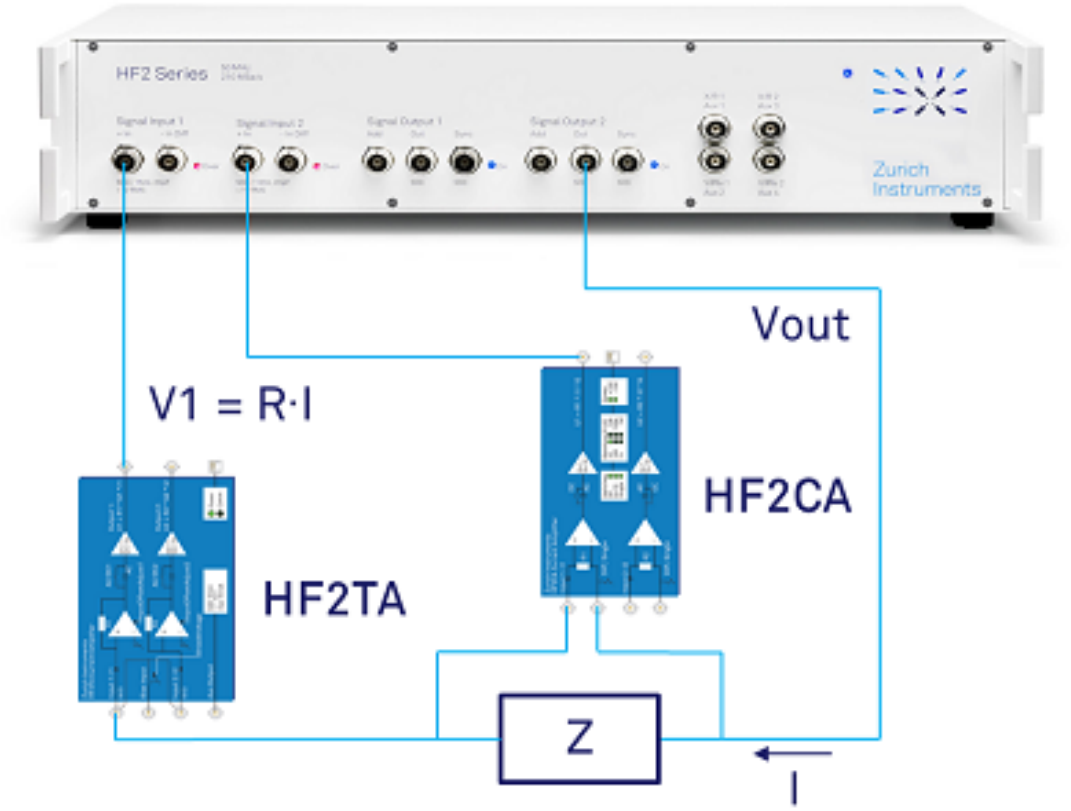


Figure 12.8. HF2IS four-point measurement setup

Note that the voltage measurement is made differentially through HF2CA then converted to single-ended input to the HF2IS while the current measurement remains single-ended throughout the current measurement path. Both HF2TA and HF2CA can be controlled using the ziControl graphical user interface. When they are connected through Ethernet cables to the back of the HF2IS instrument, ziControl will automatically add a 'Active Probes' tabs as shown in the screen shot below.

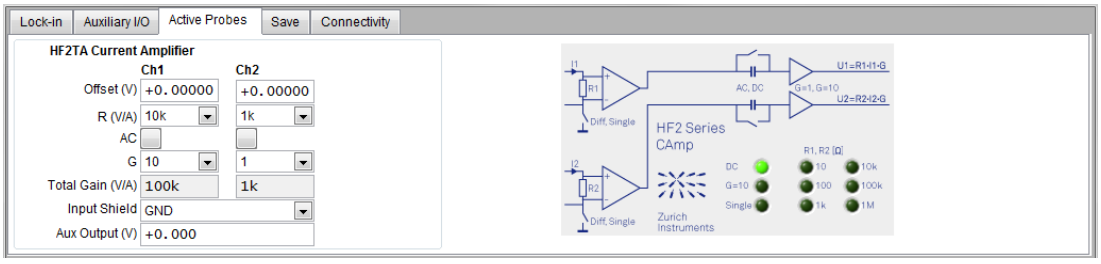


Figure 12.9. Active Probes GUI control screen

Both HF2TA and HF2CA can operate in AC-coupled or DC input. The HF2TA has a high-pass cutoff of 10 Hz while the HF2CA has a high-pass cutoff of 100 Hz. When adjusting the HF2TA transimpedance gain, the field 'Total Gain (V/A)' will display the product of R * G. It is recommended that no R1 and R2 values are selected for the HF2CA to obtain maximum input

impedance (i.e. no signal current loss through the HF2CA input ports) and therefore the most accurate current measurement.

12.4.4. Impedance Display for Solar Cell Measurement with HF2IS

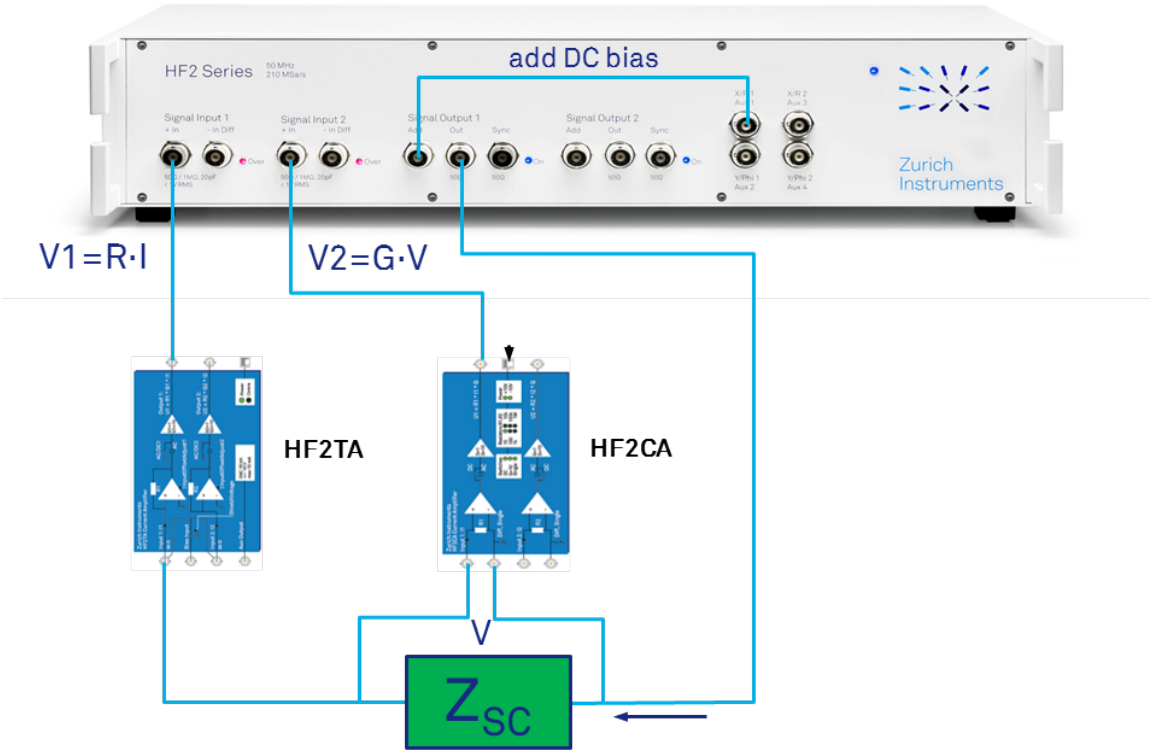


Figure 12.10. Display impedance as Cole-Cole plot

Using HF2IS in combination with HF2TA as a transimpedance amplifier and HF2CA as a voltage buffer, it is possible to characterize a small solar cell by obtaining the Cole-Cole impedance plot directly on the ziControl GUI. The maximum current that can be measured is ±100 mA which is limited by the output strength of the HF2IS output ports. The typical 4-point setup for such measurement is shown above. In addition, a DC bias can be applied to the solar cell under test through the use of one of the four HF2IS auxiliary outputs and the built-in ADD function to the output port.

On the ziControl setting shown below, HF2TA is selected for Input 1, 4-Term Z is selected for Demodulators and if a DC bias is applied, Add should be clicked for Output 1. The Sweeper can then be used to obtain the frequency response of the solar cell under test. Please note that HF2CA is not selected for Input 2. This is because HF2CA is used as a voltage buffer with ±10 V range in order to extend the HF2IS input range (±3.5 V). Since HF2CA has a gain of 1 by default, no special care is required on the ziControl setting. However, if a gain is applied on HF2CA or another 3rd-party buffer, then the Scaling function should be used to normalize the voltage measurement.

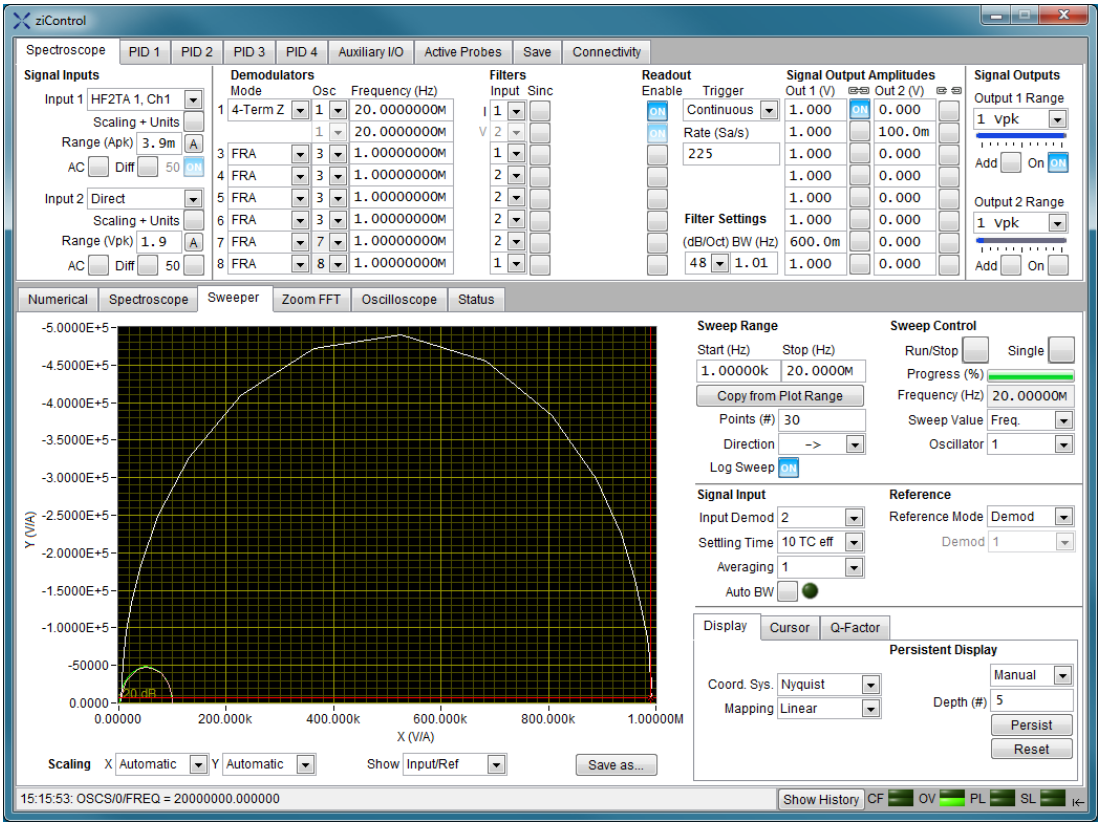


Figure 12.11. HF2IS four-point measurement setup

To observe the sweep result in Cole-Cole plot, one has to simply select Impedance and Nyquist as the display format in the Sweeper. Please ensure that Mapping is set to Linear. Since the Y axis represents the imaginary impedance, the fact that a solar cell is capacitive means that the plotted result is flipped (upside down) initially compared to the typical Cole-Cole plot which plots the imaginary impedance in absolute values. To rectify this display issue, simply choose Scaling Y to manual and set the minimum value of the Y to 0 manually. The plot should now display in the traditional Cole-Cole plot convention.

Lastly, the sweeper plot in ziControl allows user to display the previous measurement simultaneously with the current measurement. This is called Persistent Display in the Sweeper. When one wants to save a measurement plot, he or she has to only press on the Persist button. The ensuing measurement will be plotted alongside with the 'persisted' plots. The number of persistent display can be set inside the Depth field.

12.5. Performance Tests

In this section two tests are described that can be used to quickly check the DC leakage and the AC noise of the HF2TA. They can be performed by the user to do a sanity check on the validity of the measurement with HF2TA.

Table 12.5. Necessary equipment

Required equipment	Specifications	Recommended equipment
HF2 Instrument	No additional installation options required	HF2LI or HF2IS
HF2TA Current Amplifier	HF2TA Specifications	HF2TA
Digital multimeter	0.1 mV Resolution, 20 V range	Agilent 34410A
SMA to BNC cables	2 x 50 Ω, male-to-male connectors	supplied by Zurich Instruments
Ethernet cable	Category 5 or 6	supplied by Zurich Instruments

The following conditions have to be fulfilled:

1. The test equipment must be connect to the same AC power circuit. If you are unsure of the AC power circuit distribution, use a common power strip and connect all test equipment into it. Connecting the test equipment into separate AC power circuits can result in offset voltages between equipment, which can invalidate the verification test.
2. For accurate results, allow the test equipment to warm up for at least 30 minutes.
3. The HF2 Instrument as well as the HF2TA transimpedance amplifier can be controlled by means of the ziControl. Please make sure that the latest version of the LabOne and ziControl packages have been installed on host computer. More information about installing software can be found in the Chapter 1 . The ziServer installed by the LabOne software package must be running on mentioned computer. How to check if ziServer is running is described in the Chapter 6 .

The HF2TA transimpedance amplifier has 2 analog input channels, 2 analog output channels and 1 external bias input and 1 auxiliary output. For the purpose of the following tests, the external bias input will not be used. The test setup for one channel is equally valid for the other channel.

12.5.1. Input Leakage Test

Definition

This test measures the DC input leakage current of the HF2TA.

Setup

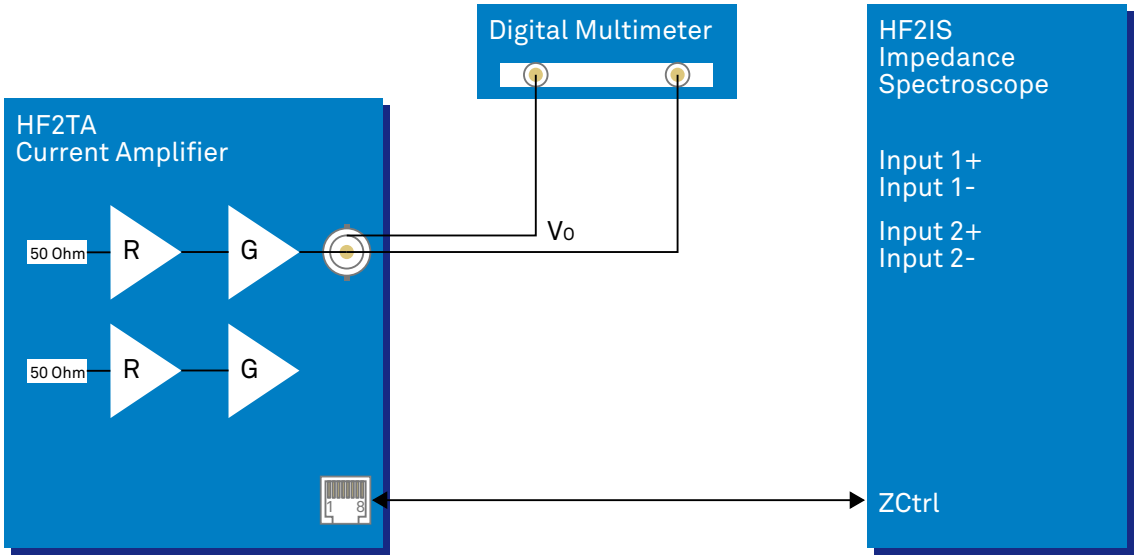


Figure 12.12. HF2TA DC input leakage measurement setup

HF2TA is connected to the HF2 Instrument via the Ethernet cable for the purpose of configuring the HF2TA. The HF2 Instrument is not used for the measurement.

Table 12.6. HF2TA settings

Ch1 Offset (V)	0.0 V
Ch1 R (V/A)	1k and 100M
Ch1 AC	OFF
G	1
Input Shield	GND
Aux Output (V)	0.0 V

Measurement

The DC leakage current can be estimated by subtracting the inherent DC offset V_{O1} of the amplifier from the total offset V_{O2} due to both the internal offset and the input leakage. For this test the input of the HF2TA is left open. Then the output is measured with a digital multimeter as shown . The input offset V_{O1} can be estimated by setting the transimpedance resistor R to 1k. The sum of the input offset plus leakage V_{O2} can be estimated by setting the transimpedance resistor R to 100M. Then, the approximate leakage can be found by:

$$I_{\text{leakage}} = \frac{|V_{O2} - V_{O1}|}{100\ \text{M}\Omega} \quad (12.1)$$

12.5.2. Input Noise Test

Definition

The noise generated by the HF2TA transimpedance amplifier itself can be expressed as input referred current noise. The following setup description enables users to verify through measurement if their HF2TA units have indeed the same noise level as specified in Table 12.4

Setup

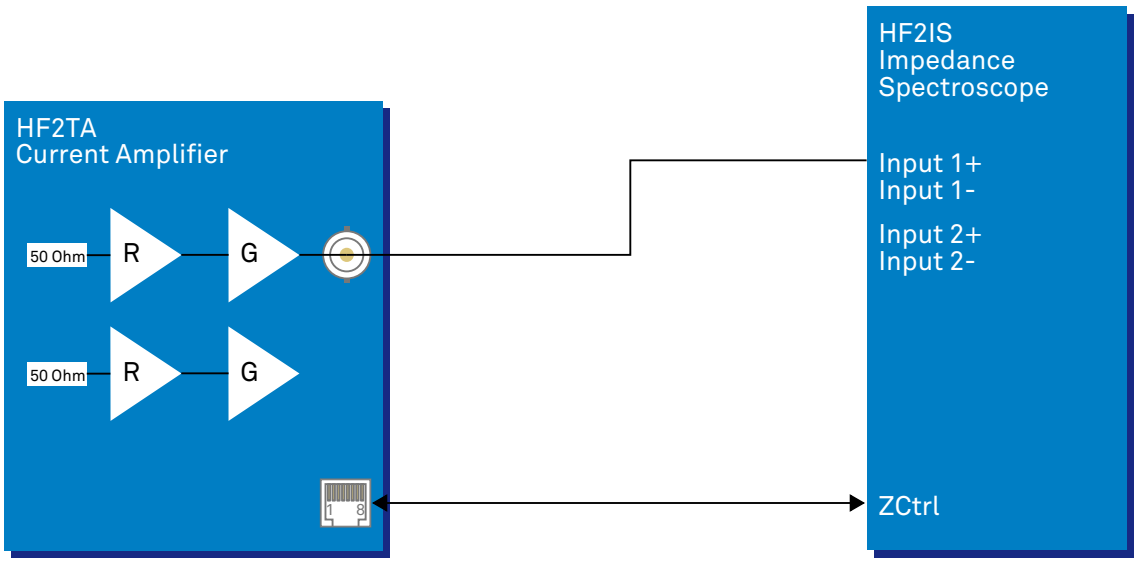


Figure 12.13. HF2TA equivalent input current noise

For this test the HF2TA transimpedance amplifier input is left open. The goal is to refer the total contribution of out noise from the amplifier itself to the input and not from any other external circuits. Since the input of the HF2TA is left open, it is only necessary to define the sweep range in the HF2 instrument since no drive voltage is required.

The HF2 instrument settings for the test are given in the table below.

Table 12.7. HF2 instrument settings

Ch1 Signal Inputs Range	press 'A' for auto range
Ch1 Signal Inputs AC/ Diff/ 50	ON
Ch1 Scale	1/R
Filter BW setting type	BW NEP (noise equivalent power BW)
Filter BW	1 Hz

Notice that Scale has been set to 1/R where R is the HF2TA transimpedance value. This is to obtain the noise current referred back to input.

Measurement

The Sweeper tool will be used to get the measurement result. Be sure to turn off Normalize since we are interested in the absolute value. The settings which have to be set in the Sweeper tool section are following:

Table 12.8. Frequency sweeper settings

Sweep Range Start	1 kHz
Sweep Range Stop	50 MHz
Sweep Range Points	50
Sweep Range Log Sweep	ON
Filter Auto BW	OFF
Filter Precision	3 TC
Filter Averaging	32
Display	Input Only
Display Mapping	linear
Coordinate System	Polar
Result Unit	$V_{RMS}/\sqrt{Hz}$

After settings have been applied, to start the measurement one should run continuous sweep. The input referred current noise is directly displayed in the Sweeper Tool over the specified frequency. The result can then be compared to the input referred noise table.

12.6. Cable Recommendation

Table 12.9. HF2TA cable recommendation

Function	Connector / cable type	Vendor / part number
SMA to BNC connection		
SMA to BNC cable	BNC jack to SMA plug	Digikey J3606-ND
SMA to BNC adapter	BNC jack to SMA plug	Digikey J10098-ND
		Farnell / Newark 4195930
	BNC plug to SMB plug	Farnell / Newark 1654647
		Digikey ACX1324-ND
Custom access or cable assembly		
Cable	Cable type RG-174	Digikey A307-100-ND
		Farnell / Newark 1387745
SMA to cable	SMA plug to RG-174 cable	Digikey A32326-ND
		Farnell /Newark 2112459
BNC to cable	BNC plug to RG-174 cable	Tyco Electronics 1-5227079-6
		Digikey A32212-ND
		Farnell / Newark 1831701

Glossary

This glossary provides easy to understand descriptions for many terms related to measurement instrumentation including the abbreviations used inside this user manual.

A

A/D	Analog to Digital See Also ADC.
AC	Alternate Current
ADC	Analog to Digital Converter
AM	Amplitude Modulation
Amplitude Modulated AFM (AM-AFM)	AFM mode where the amplitude change between drive and measured signal encodes the topography or the measured AFM variable. See Also Atomic Force Microscope.
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
Atomic Force Microscope (AFM)	Microscope that scans surfaces by means an oscillating mechanical structure (e.g. cantilever, tuning fork) whose oscillating tip gets so close to the surface to enter in interaction because of electrostatic, chemical, magnetic or other forces. With an AFM it is possible to produce images with atomic resolution. See Also Amplitude Modulated AFM, Frequency Modulated AFM, Phase modulation AFM.
AVAR	Allen Variance

B

Bandwidth (BW)	The signal bandwidth represents the highest frequency components of interest in a signal. For filters the signal bandwidth is the cut-off point, where the transfer function of a system shows 3 dB attenuation versus DC. In this context the bandwidth is a synonym of cut-off frequency $f_{cut-off}$ or 3dB frequency f_{-3dB}. The concept of bandwidth is used when the dynamic behavior of a signal is important or separation of different signals is required. In the context of a open-loop or closed-loop system, the bandwidth can be used to indicate the fastest speed of the system, or the highest signal update change rate that is possible with the system. Sometimes the term bandwidth is erroneously used as synonym of frequency range. See Also Noise Equivalent Power Bandwidth.
BNC	Bayonet Neill-Concelman Connector

C

CF	Clock Fail (internal processor clock missing)
Common Mode Rejection Ratio (CMRR)	Specification of a differential amplifier (or other device) indicating the ability of an amplifier to obtain the difference between two inputs while rejecting the components that do not differ from the signal (common mode). A high CMRR is important in applications where the signal of interest is represented by a small voltage fluctuation superimposed on a (possibly large) voltage offset, or when relevant information is contained in the voltage difference between two signals. The simplest mathematical definition of common-mode rejection ratio is: CMRR = 20 * log(differential gain / common mode gain).
CSV	Comma Separated Values

D

D/A	Digital to Analog
DAC	Digital to Analog Converter
DC	Direct Current
DDS	Direct Digital Synthesis
DHCP	Dynamic Host Configuration Protocol
DIO	Digital Input/Output
DNS	Domain Name Server
DSP	Digital Signal Processor
DUT	Device Under Test
Dynamic Reserve (DR)	The measure of a lock-in amplifier's capability to withstand the disturbing signals and noise at non-reference frequencies, while maintaining the specified measurement accuracy within the signal bandwidth.

E

XML	Extensible Markup Language. See Also XML.

F

FFT	Fast Fourier Transform
FIFO	First In First Out
FM	Frequency Modulation
Frequency Accuracy (FA)	Measure of an instrument's ability to faithfully indicate the correct frequency versus a traceable standard.
Frequency Modulated AFM (FM-AFM)	AFM mode where the frequency change between drive and measured signal encodes the topography or the measured AFM variable. See Also Atomic Force Microscope.

Frequency Response Analyzer (FRA)	Instrument capable to stimulate a device under test and plot the frequency response over a selectable frequency range with a fine granularity.
Frequency Sweeper	See Also Frequency Response Analyzer.

G

Gain Phase Meter	See Also Vector Network Analyzer.
GPIB	General Purpose Interface Bus
GUI	Graphical User Interface

I

I/O	Input / Output
Impedance Spectroscope (IS)	Instrument suited to stimulate a device under test and to measure the impedance (by means of a current measurement) at a selectable frequency and its amplitude and phase change over time. The output is both amplitude and phase information referred to the stimulus signal.
Input Amplitude Accuracy (IAA)	Measure of instrument's capability to faithfully indicate the signal amplitude at the input channel versus a traceable standard.
Input voltage noise (IVN)	Total noise generated by the instrument and referred to the signal input, thus expressed as additional source of noise for the measured signal.
IP	Internet Protocol

L

LAN	Local Area Network
LED	Light Emitting Diode
Lock-in Amplifier (LI, LIA)	Instrument suited for the acquisition of small signals in noisy environments, or quickly changing signal with good signal to noise ratio - lock-in amplifiers recover the signal of interest knowing the frequency of the signal by demodulation with the suited reference frequency - the result of the demodulation are amplitude and phase of the signal compared to the reference: these are value pairs in the complex plane (X, Y), (R, Θ).

M

Media Access Control address (MAC address)	Refers to the unique identifier assigned to network adapters for physical network communication.
Multi-frequency (MF)	Refers to the simultaneous measurement of signals modulated at arbitrary frequencies. The objective of multi-frequency is to increase the information that can be derived from a measurement which is particularly important for one-time, non-repeating events, and to increase the speed of a measurement since different frequencies do not have to be applied one after the other. See Also Multi-harmonic.

Multi-harmonic (MH)	Refers to the simultaneous measurement of modulated signals at various harmonic frequencies. The objective of multi-frequency is to increase the information that can be derived from a measurement which is particularly important for one-time, non-repeating events, and to increase the speed of a measurement since different frequencies do not have to be applied one after the other. See Also Multi-frequency.

N

Noise Equivalent Power Bandwidth (NEPBW)	Effective bandwidth considering the area below the transfer function of a low-pass filter in the frequency spectrum. NEPBW is used when the amount of power within a certain bandwidth is important, such as noise measurements. This unit corresponds to a perfect filter with infinite steepness at the equivalent frequency. See Also Bandwidth.
Nyquist Frequency (NF)	For sampled analog signals, the Nyquist frequency corresponds to two times the highest frequency component that is being correctly represented after the signal conversion.

O

Output Amplitude Accuracy (OAA)	Measure of an instrument's ability to faithfully output a set voltage at a given frequency versus a traceable standard.
OV	Over Volt (signal input saturation and clipping of signal)

P

PC	Personal Computer
PD	Phase Detector
Phase-locked Loop (PLL)	Electronic circuit that serves to track and control a defined frequency. For this purpose a copy of the external signal is generated such that it is in phase with the original signal, but with usually better spectral characteristics. It can act as frequency stabilization, frequency multiplication, or as frequency recovery. In both analog and digital implementations it consists of a phase detector, a loop filter, a controller, and an oscillator.
Phase modulation AFM (PM-AFM)	AFM mode where the phase between drive and measured signal encodes the topography or the measured AFM variable. See Also Atomic Force Microscope.
PID	Proportional-Integral-Derivative
PL	Packet Loss (loss of packets of data between the instruments and the host computer)

R

RISC	Reduced Instruction Set Computer
Root Mean Square (RMS)	Statistical measure of the magnitude of a varying quantity. It is especially useful when variates are positive and negative, e.g., sinusoids, sawtooth,

	square waves. For a sine wave the following relation holds between the amplitude and the RMS value: $U_{RMS} = U_{PK} / \sqrt{2} = U_{PK} / 1.41$. The RMS is also called quadratic mean.
RT	Real-time

S

Scalar Network Analyzer (SNA)	Instrument that measures the voltage of an analog input signal providing just the amplitude (gain) information. See Also Spectrum Analyzer, Vector Network Analyzer.
SL	Sample Loss (loss of samples between the instrument and the host computer)
Spectrum Analyzer (SA)	Instrument that measures the voltage of an analog input signal providing just the amplitude (gain) information over a defined spectrum. See Also Scalar Network Analyzer.
SSH	Secure Shell

T

TC	Time Constant
TCP/IP	Transmission Control Protocol / Internet Protocol
Thread	An independent sequence of instructions to be executed by a processor.
Total Harmonic Distortion (THD)	Measure of the non-linearity of signal channels (input and output)
TTL	Transistor to Transistor Logic level

U

UHF	Ultra-High Frequency
UHS	Ultra-High Stability
USB	Universal Serial Bus

V

VCO	Voltage Controlled Oscillator
Vector Network Analyzer (VNA)	Instrument that measures the network parameters of electrical networks, commonly expressed as s-parameters. For this purpose it measures the voltage of an input signal providing both amplitude (gain) and phase information. For this characteristic an older name was gain phase meter. See Also Gain Phase Meter, Scalar Network Analyzer.

X

XML	Extensible Markup Language: Markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

Z

ZCtrl	Zurich Instruments Control bus
ZoomFFT	This technique performs FFT processing on demodulated samples, for instance after a lock-in amplifier. Since the resolution of an FFT depends on the number of point acquired and the spanned time (not the sample rate), it is possible to obtain very highly resolution spectral analysis.
ZSync	Zurich Instruments Synchronization bus

Index

A

Active Probes
  Tab, 137, 178
AM (see Modulation, Amplitude)
Attenuation, 43
Auto range, 40
Auxiliary, 311
  Tab, 130, 171
    Tutorial, 46

B

Bandwidth, 320, 353
  Demodulator, 310
  Filter (see Filter, Bandwidth)
  Limit, 42
  PLL (see PLL, Bandwidth)

C

CA (see Preamplifier, CA)
Calibration, 10
CMRR, 308, 319, 337
  Definition, 337
Computer requirements, 306
Connectivity
  Tab, 135, 176
    Tutorial, 47
Coupling
  AC, 40, 105, 110
  DC, 105, 110
Cut-off frequency, 353

D

Damage threshold, 306
Diff (see Differential mode)
Differential mode, 40
DIO, 309
Dynamic reserve, 308

F

Filter, 355
  Bandwidth, 169
  Compensation, 362
  Settings, 59
  Settling time, 355- 356
  Sinc, 310, 359
Filter slope, 51, 57
FM (see Modulation, Frequency)
Frequency
  Accuracy, 343
  Range, 308
    External reference, 308
    Internal reference, 308
  Resolution, 308
Full range sensitivity, 308, 357- 358

H

Harmonic
  Distortion, 317, 339
    Definition, 339
  Rejection, 310
HF input noise (see Noise, HF Input)
HF signal inputs (see Input, HF)
HF signal outputs (see Output, HF)
HF2 back panel, 30
HF2 front panel, 28
HF2 functional diagram, 25

I

Impedance, 143, 184
  Input, 40
  Spectroscope, 165
Input
  Amplitude Accuracy, 346
  HF, 308
Input range
  AC coupling, 308
  DC coupling, 308
  settings, 308
Installation
  Linux, 17
  Microsoft .NET Framework, 14
  Windows, 12

L

LabOneSoftware package, 12
LabVIEW, 16
  API, 205
Linux
  Software installation, 17
Lock-in
  Principal, 350
  Tab, 104
    Tutorial, 39
Log files, 22, 159, 200
  History log, 158, 199

M

Microsoft .NET Framework, 14
Modulation
  Amplitude, 67- 68, 114, 140
    Tutorial, 67
  Frequency, 72- 74, 76, 114, 140
    Tutorial, 72
  Tab, 113
  Tools
    Numerical, 140
Multi-frequency, 36
  Tab, 39, 108

N

NEPBW, 44, 149, 189, 353
Node
  Concept, 207
  Detailed node description, 229
  Hierarchical overview, 222
  Leaf, 207
  Node hierarchy, 206
Nodes
  1V2, 240
  1V8, 240
  2V5, 240
  3V3, 241
  5V0, 241
  ABOUT, 229
  AC, 263
  ACTIVE, 261
  ACTIVETHRESHOLD, 262
  ACTIVETIMECONSTANT, 261
  ADC0MAX, 238
  ADC0MIN, 238
  ADC1MAX, 238
  ADC1MIN, 238
  ADCCLIP, 236
  ADCSELECT, 242, 248
  ADCTHRESHOLD, 252
  ADD, 265
  AMPLITUDES, 266
  AUTOCENTER, 248
  AUTOPID, 251
  AUTOTIMECONSTANT, 250
  AUXAVG, 252
  AUXINS, 273
  AUXOUTS, 274
  AVERAGING, 273
  BIASOUT, 279
  BINARY, 234
  BWLIMIT, 269
  BYTESRECEIVED, 239
  BYTESSENT, 239
  CAMP, 277
  CENTER, 259
  CHANNEL, 268
  CLOCKBASE, 231, 231
  CODE, 233
  CONFIG, 230
  COPYRIGHT, 230
  CPUS, 276
  CURRENTGAIN, 280
  D, 257
  DC, 278, 280
  DCMLOCK, 235
  DECIMATION, 271
  DEMODS, 241
  DEMODSAMPLELOSS, 237
  DEMODSELECT, 275
  DEV, 231
  DEVTYPE, 233
  DIFF, 264
  DIOS, 271
  DRIVE, 271
  DTIMECONSTANT, 251
  ENABLE, 249, 260, 261
  ENABLES, 266
  ERROR, 250, 259
  EXTBIAS, 279
  EXTCLK, 232, 271
  FEATURES, 232
  FIFOLEVEL, 237
  FLAGS, 234
  FMCOEFF, 246
  FREQ, 246
  FREQCENTER, 249
  FREQDELTA, 252
  FREQRANGE, 249
  FWREVISION, 229
  FX2RX, 235
  GAIN, 278
  HARMONIC, 244, 253
  HWREVISION, 232
  I, 257
  IMP50, 264
  INACTIVETHRESHOLD, 263
  INACTIVETIMECONSTANT, 262
  INPUT, 254, 273
  INPUTCHANNEL, 255
  ITIMECONSTANT, 251
  MEANMSGCNT, 240
  MEANPOLLCNT, 239
  MIXERCLIP, 236
  MODE, 247
  MODS, 246
  MONITOROFFSET, 258
  MONITORSCALE, 258
  OFFSET, 275, 281
  ON, 265
  OPEN, 230
  OPTIONS, 233
  ORDER, 242, 253
  OSCS, 245
  OSCSELECT, 244
  OUTPUT, 255, 272, 276
  OUTPUTCHANNEL, 255
  OUTPUTDEFAULT, 256
  OUTPUTDEFAULTENABLE, 256
  OUTPUTSELECT, 274
  P, 256
  PGAIN, 251
  PHASESHIFT, 244
  PHYSICAL, 240
  PIDS, 254
  PKGLOSS, 235
  PLL, 261

PLLLOCK, 234
PLLS, 248
PORT, 230
PROGRAM, 276
R, 277
RANGE, 259, 263, 265
RATE, 243, 260
REVISION, 229
SAMPLE, 245, 247, 273
SCALE, 275
SCOPES, 267
SCOPESKIPPED, 237
SERIAL, 232
SETPOINT, 253, 257
SETPOINTSELECT, 258
SHIFT, 260
SIDEBAND, 247
SIGINS, 263
SIGOUTS, 264
SINC, 245
SINGLEENDED, 279
STATS, 239
STATUS, 233
SYNCSELECT0, 272
SYNCSELECT1, 272
SYSTEM, 231
TAMP, 279
TEMP, 241
TIME, 233, 270
TIMECONSTANT, 243, 250
TIPPROTECT, 260
TREES, 230
TRIGCHANNEL, 268
TRIGEDGE, 269
TRIGGER, 243
TRIGHOLDOFF, 270
TRIGLEVEL, 269
USERREGS, 277
VALUE, 274
VERSION, 229
VOLTAGEGAIN, 281
WAVE, 270
WAVEFORMS, 267
WORKLOAD, 276
ZCTRLS, 277
ZI, 229
Noise, 331
1/f, 351
Definition, 148, 189, 331
HF Input, 315
Phase noise, 310
SNR
Definition, 148, 189
Tutorial, 66- 66
Numerical
Tool, 139, 180
Tutorial, 41
Nyquist sampling theorem, 40, 307

O

Oscilloscope
Analysis tab, 157, 198
FFT tab, 155, 196
Histogram tab, 158, 199
Tool, 152, 193
Tutorial, 42
Output
ADD, 55, 310
Amplitude Accuracy, 344
HF, 310
Sync, 28

P

Performance diagrams, 315
Phase
Resolution, 308
PID
control units, 127
Settings, 121, 127
Auto tune, 87, 91, 126
Input, 123
Output, 124
Parameter, 125
Tab, 121
Tutorial, 87, 91
Ziegler-Nichols Method, 91
PLL, 79
Bandwidth, 79
Tutorial, 79, 84
PLL (Phase-locked Loop)
Advisor, 119
Tab, 116
Polling Data
Concept, 207
Preamplifier
CA, 137, 178
TA, 138, 179

Q

Quadrature, 350

R

Real-time
Tab, 129, 170
Reference mode
Auto, 36, 105
External, 34, 105, 308
Tutorial, 60- 64, 61
Internal, 33, 105, 308
Reference signal, 308, 350
Reference source, 40
External, 40
Internal, 40

RMS value, 351

S

Save
  Tab, 133, 174
    Tutorial, 46
Sensitivity (see Full range sensitivity)
Software Installation, 12
  Linux, 17
  Microsoft .NET Framework, 14
  Requirements, Linux, 17
  Requirements, Windows, 12
  Supported versions of Linux, 17
  Supported versions of Windows, 12
  Windows, 12
Spectroscope
  IS Tab, 165
  Tool, 140, 181
    Tutorial, 41
Stability
  Input amplitude, 308
Status
  Tool, 158, 199
    Tutorial, 45
Sweeper
  Tool, 142, 183
    Tutorial, 43

T

TA (see Preamplifier, TA)
Time constant, 310
Transfer function, 43

U

UHS, 311

W

Windows
  Software installation, 12

Z

ZCtrl, 27
ziControl, 39
  Installation, 16
  Software package, 12
Ziegler-Nichols PID tuning method, 91
ziServer, 38, 204, 218
  Check status, 206
  Installer issue with 64-bit Windows version, 16
  on Linux, 211
  on Windows, 209
  ziService (Linux), 17
ziService, 38, 206
ziService Program, 17
ZoomFFT, 362- 363
  Tool, 147, 188
    Tutorial, 44
ZSync, 27